9-10-2019

Big Data Analytics for Complex Systems

Ashraf Mohamed Abou Tabl
University of Windsor

Big Data Analytics for Complex Systems

by

Ashraf Mohamed Abou Tabl

A Dissertation
Submitted to the Faculty of Graduate Studies
through the Industrial and Manufacturing Systems Engineering Graduate
Program in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy at the
University of Windsor

Windsor, Ontario, Canada

Big Data Analytics for Complex Systems

by

Ashraf Mohamed Abou Tabl

APPROVED BY:

_____

Shihab S. Asfour, External Examiner
University of Miami

_____

A. Hussein
Department of Mathematics and Statistics

_____

Z. J. Pasek
Department of Mechanical, Automotive and Materials Engineering

_____

H. ElMaraghy
Department of Mechanical, Automotive and Materials Engineering

_____

W. ElMaraghy, Advisor
Department of Mechanical, Automotive and Materials Engineering

September 10, 2019

DECLARATION OF CO-AUTHORSHIP / PREVIOUS PUBLICATION

I. Co-Authorship

I hereby declare that this thesis incorporates material that is a result of joint research of the author and his supervisor Prof. Waguih ElMaraghy. Chapter 5 of the thesis was co-authored with Professor Alioune Ngom, Professor Luis Rueda, Dr. Abedalrhman Alkhateeb, Mr. P. Quang, and Mr. S. Jubair. In all cases, the key ideas, primary contributions, experimental designs, data analysis, interpretation, and writing were performed by the author, and the contribution of the co-authors was providing feedback on refinement of ideas and editing of the manuscript. This joint research has been submitted to Journals and Conferences that are listed below.

I am aware of the University of Windsor Senate Policy on Authorship, and I certify that I have properly acknowledged the contribution of other researchers to my thesis, and have obtained written permission from each of the co-authors to include the above material(s) in my thesis.

I certify that, with the above qualification, this thesis, and the research to which it refers, is the product of my own work.

II. Declaration of Previous Publication

This thesis includes seven original papers that have been previously published/submitted for publication in peer-reviewed journals and conferences, as follows:

Thesis Chapter	Publication title/full citation	Publication status*
4	Ashraf Abou Tabl, W. ElMaraghy. "Big Data Analytics for Defects Detection in Manufacturing Systems (Industry 4.0)", International Journal of Computer Integrated Manufacturing(IJCIM) ID: TCIM-2019-IJCIM-0178.	Journal (Submitted)

5	Ashraf Abou Tabl, A. Alkhateeb, L. Rueda, W. ElMaraghy, A. Ngom. "A machine learning approach for identifying gene biomarkers guiding the treatment of breast cancer," Frontiers in Genetics. doi: 10.3389/fgene.2019.00256.	Journal (published)
5	Ashraf Abou Tabl, A. Alkhateeb, P. Quang, L. Rueda, W. ElMaraghy, A. Ngom." A novel approach for identifying relevant genes for breast cancer survivability on specific therapies" Evolutionary Bioinformatics, 2018, Evolutionary Bioinformatics, 14, doi: 10.1177/1176934318790266.	Journal (published)
5	S. Jubair, A. Alkhateeb, A Abou Tabl, L. Rueda, A. Ngom, "Identifying subtype specific network-biomarkers of breast cancer survivability", Evolutionary Bioinformatics, (EVB-2019-0142).	Journal (Submitted)
5	Ashraf Abou Tabl, Alkhateeb A, ElMaraghy W and Ngom A. Machine learning model for identifying gene biomarkers for breast cancer treatment survival [version 1; not peer reviewed]. F1000Research 2017, 6(ISCB Comm J):1681 (doi: 10.7490/f1000research.1114873.1) (Non-referee Journal) (Abstract and Poster)	Journal (published)
5	Ashraf Abou Tabl, Alkhateeb, A., ElMaraghy, W., & Ngom, A. (2017, August). Machine Learning Model for Identifying Gene Biomarkers for Breast Cancer Treatment Survival. In Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics (pp. 607-607) (ACM 2017), Boston, MA. doi>10.1145/3107411.3108217	Conference proceeding (published)
5	Ashraf Abou Tabl, Alkhateeb, A., Rueda, L., ElMaraghy, W., & Ngom, A. (2018, March). Identifying gene biomarkers for breast cancer survival using a tree-based approach. BHI 2018 IEEE International Conference on Biomedical and Health Informatics, Las Vegas, NV, USA, March 2018.	Conference proceeding (published)

I certify that I have obtained written permission from the copyright owner(s) to include the above-published material(s) in my thesis. I certify that the above material describes work completed during my registration as a graduate student at the University of Windsor.

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained written permission from the copyright owner(s) to include such material(s) in my thesis.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

The evolution of technology in all fields led to the generation of vast amounts of data by modern systems. Using data to extract information, make predictions, and make decisions is the current trend in artificial intelligence. The advancement of big data analytics tools made accessing and storing data easier and faster than ever, and machine learning algorithms help to identify patterns in and extract information from data. The current tools and machines in health, computer technologies, and manufacturing can generate massive raw data about their products or samples. The author of this work proposes a modern integrative system that can utilize big data analytics, machine learning, super-computer resources, and industrial health machines' measurements to build a smart system that can mimic the human intelligence skills of observations, detection, prediction, and decision-making. The applications of the proposed smart systems are included as case studies to highlight the contributions of each system.

The first contribution is the ability to utilize big data revolutionary and deep learning technologies on production lines to diagnose incidents and take proper action. In the current digital transformational industrial era, Industry 4.0 has been receiving researcher attention because it can be used to automate production-line decisions. Reconfigurable manufacturing systems (RMS) have been widely used to reduce the setup cost of restructuring production lines. However, the current RMS modules are not linked to the cloud for online decision-making to take the proper decision; these modules must connect to an online server (super-computer) that has big data analytics and machine learning capabilities. The online means that data is centralized on cloud (supercomputer) and accessible in real-time. In this study, deep neural networks are utilized to detect the decisive features of a product and build a prediction model in which the iFactory will make the necessary decision for the defective products. The Spark ecosystem is used to manage the access, processing, and storing of the big data streaming. This contribution is implemented as a closed cycle, which for the best of our knowledge, no one in the literature has introduced big data analysis using deep learning on real-time applications in the manufacturing system. The code shows a high accuracy of 97% for classifying the normal versus defective items.

The second contribution, which is in Bioinformatics, is the ability to build supervised machine learning approaches based on the gene expression of patients to predict proper treatment for breast cancer. In the trial, to personalize treatment, the machine learns the genes that are active in the patient cohort with a five-year survival period. The initial condition here is that each group must only undergo one specific treatment. After learning about each group (or class), the machine can personalize the treatment of a new patient by diagnosing the patients' gene expression. The proposed model will help in the diagnosis and treatment of the patient. The future work in this area involves building a protein-protein interaction network with the selected genes for each treatment to first analyze the motives of the genes and target

them with the proper drug molecules. In the learning phase, a couple of feature-selection techniques and supervised standard classifiers are used to build the prediction model. Most of the nodes show a high-performance measurement where accuracy, sensitivity, specificity, and F-measure ranges around 100%.

The third contribution is the ability to build semi-supervised learning for the breast cancer survival treatment that advances the second contribution. By understanding the relations between the classes, we can design the machine learning phase based on the similarities between classes. In the proposed research, the researcher used the Euclidean matrix distance among each survival treatment class to build the hierarchical learning model. The distance information that is learned through a non-supervised approach can help the prediction model to select the classes that are away from each other to maximize the distance between classes and gain wider class groups. The performance measurement of this approach shows a slight improvement from the second model. However, this model reduced the number of discriminative genes from 47 to 37. The model in the second contribution studies each class individually while this model focuses on the relationships between the classes and uses this information in the learning phase. Hierarchical clustering is completed to draw the borders between groups of classes before building the classification models. Several distance measurements are tested to identify the best linkages between classes. Most of the nodes show a high-performance measurement where accuracy, sensitivity, specificity, and F-measure ranges from 90% to 100%.

All the case study models showed high-performance measurements in the prediction phase. These modern models can be replicated for different problems within different domains. The comprehensive models of the newer technologies are reconfigurable and modular; any newer learning phase can be plugged-in at both ends of the learning phase. Therefore, the output of the system can be an input for another learning system, and a newer feature can be added to the input to be considered for the learning phase.

DEDICATION

To God

For the privilege of giving me this life

To my Father, Mother, and Brothers

For their infinite love and support throughout my life

To my Wife and my Kids

For their love and endless joy that they brought to my life

ACKNOWLEDGMENT

Firstly, I would like to extend my sincere gratitude to my dissertation adviser, Professor Waguih ElMaraghy, for giving me the opportunity to collaborate with him and for his guidance, encouragement, time and effort throughout the course of this research. His continuous support has benefited my research tremendously.

I would like to thank the committee members for their feedback and constructive comments that led me to make significant improvements to this dissertation. I extend a special thanks to Professor Hoda ElMaraghy for her comments and suggestions during committee meetings and Intelligent Manufacturing Systems (IMS) Centre meetings. My sincere thanks also go to Professor Zbigniew J. Pasek for his guidance, suggestions, and challenging questions. Many thanks to Professor Abdulkadir Hussein for providing valuable feedback and recommendations.

I would also like to thank my current and former colleagues at the IMS Centre for their help and support; many thanks to Dr. Mohamed Hanafy, Dr. Mohamed Kashkoush, Dr. Mohamed Abbas, Dr. Abdulrahman Seleim, Dr. Jessica Olivares, Mr. Ahmed Marzouk, Mr. Mostafa Moussa, and Mr. Hamid Tabti.

I would like to express my best and sincerest gratitude to Dr. Abedalrhman Alkhateeb, Professor Alioune Ngom, and Professor Luis Rueda from the School of Computer Science for all their support, help, guidance, patience, and encouragement, as well as to my parents, family, and friends for being always there for me. None of this would have been possible without you.

Additionally, I would like to thank the Department of Mechanical, Automotive & Materials Engineering for the Graduate Assistantship it provided.

TABLE OF CONTENTS

DECLARATION OF CO-AUTHORSHIP / PREVIOUS PUBLICATION iii
ABSTRACT vi
DEDICATION viii
ACKNOWLEDGMENT ix
LIST OF FIGURES xiii
LIST OF TABLES xvi
LIST OF ABBREVIATIONS xvii
CHAPTER 1. INTRODUCTION 1
1.1 Overview 1
1.2 Research Motivation 3
1.3 Problem Statement 4
1.4 Research Objective and Scope 5
1.5 Research Gaps and Novelty 8
1.6 Thesis Hypothesis 10
CHAPTER 2. LITERATURE REVIEW 12
2.1 Overview 12
2.2 Big Data Analytics in Complex Systems-Related Domains 12
2.3 Data Mining, Artificial Intelligence, Machine Learning, and Neural Networks (Deep Learning) 16
2.3.1 Data Mining 17
2.3.2 Artificial Intelligence 18
2.3.3 Machine Learning (ML) 19
2.4 Neural Networks 20
2.5 Manufacturing Systems Paradigms 24
2.6 Cyber-Physical Systems (CPSs) 26
2.7 Internet of Things (IoT) 27
2.8 Cloud Computing (CC) 28
2.9 Big Data Analytics for Healthcare 30
CHAPTER 3. RESEARCH METHODOLOGY 31
3.1 Overview 31
3.2 Research Plan 31
3.3 Manufacturing System 35
3.3.1 Big Data Predictive Analytics Workflow 35
3.3.2 IDEF0 for Manufacturing System 36

3.3.3 Convolutional Neural Network (CNN) 41
3.4 Healthcare 42
3.4.1 IDEF0 for Healthcare 42
3.4.2 Naive Bayes classifier 44
3.4.3 Support Vector Machine (SVM) 44
CHAPTER 4. MANUFACTURING SYSTEMS APPLICATION 46
4.1 Overview 46
4.2 Case Study 1 48
Big Data Analytics for Defects Detection in Manufacturing Systems (Industry 4.0). (Abou Tabl et al. 2019) 48
4.2.1 Abstract 48
4.2.2 Problem Definition 48
4.2.3 Introduction 49
4.2.4 Background 51
4.2.5 Materials and Methods 53
4.2.6 Model Architecture - Methodologies and Model Design 57
4.2.7 Tuning the Model Hyper parameters 60
4.2.8 Results 60
4.2.9 Conclusion 62
CHAPTER 5. HEALTH INFORMATICS APPLICATION 63
5.1 Overview 63
5.2 Case Study 2 64
A machine learning approach for identifying gene biomarkers guiding the treatment of breast cancer. (Tabl et al. 2019) 64
5.2.1 Abstract—Objective 64
5.2.2 Background 64
5.2.3 Materials and Methods 66
5.2.4 Results and Discussion 69
5.2.5 Biological Insight 72
5.2.6 Conclusion 80
5.3 Case Study 3 81
A novel approach for identifying relevant genes for breast cancer survivability on specific therapies. (Tabl et al. 2018) 81
5.3.1 Abstract 81
5.3.2 Background 81
5.3.3 Materials and Methods 82

5.3.4 The bottom-up Multi-class Classification approach 85
5.3.5 Feature Selection 90
5.3.6 Class Imbalance 90
5.3.7 Classification 90
5.3.8 Results and Discussion 91
5.3.9 Biological Insight 97
5.3.10 Conclusion 99
5.3.11 Supplementary Materials 99
CHAPTER 6. CONCLUSION 105
6.1 Overview 105
6.2 Novelties and Contributions 105
6.2.1 Manufacturing Domain 105
6.2.2 Health Informatics Domain 105
6.3 Research Significance and Benefits 106
6.3.1 Manufacturing Domain 106
6.3.2 Health Informatics Domain 107
6.4 Limitations 107
6.5 Future Work 108
6.6 Conclusion 108
REFERENCES 110
APPENDIX A: SAMPLE OF PATIENTS GENE EXPRESSION AND CLINICAL DATA 122
APPENDIX B: COMPUTE CANADA SETUP AND CNN CODE FOR MANUFACTURING SYSTEM (SAMPLE) 124
A. Compute Canada Setup (Sample) 124
B. CNN Code (Sample) 124
C. CNN Testing Code (Sample) 128
D. CNN Web Cam Code (Sample) 130
E. CNN Code Results (Sample) 132
APPENDIX C: CLASS IMBALANCE TECHNIQUES FOR HEALTH INFORMATICS SYSTEM (SAMPLE) 133
A. Over-sampling with synthetic data 133
B. Using a cost-sensitive classifier 133
C. Resampling 134
VITA AUCTORIS 135

LIST OF FIGURES

Figure 1. Data, information, knowledge, and insight. 1
Figure 2. Extracting business value (fifth V) from the four main V's of big data by IBM (Steve Chadwick 2016). 3
Figure 3. Significant applications of big data. 6
Figure 4. Components of smart manufacturing (Industry 4.0). 12
Figure 5. Big Data project implementation phases (Dutta & Bose, 2015). 13
Figure 6. Advanced analytics for complex manufacturing processes (Auschitzky et al., 2014). 14
Figure 7. The framework of big data in PLM (J. Li et al., 2015). 15
Figure 8. The relationship among different AI disciplines. 16
Figure 9. Venn diagram for AI, DM, and big data. 18
Figure 10. Human neuron and A multilayer perceptron (MLP) with two hidden layers (Liu et al. 2018). 22
Figure 11. 2D data set displayed on a graph. 23
Figure 12 Manufacturing systems paradigms positioning. 25
Figure 13. Cyber-Physical System structure (Gölzer et al., 2015). 27
Figure 14. Cloud Computing ("Cloud Computing," 2016). 29
Figure 15. Research plan. 32
Figure 16. IDEF0 for big data analytics for both domains. 33
Figure 17. Detailed IDEF0 for big data analytics for both domains. 34
Figure 18. Zachman framework. 35
Figure 19. Big Data analytics predictive workflow. 36
Figure 20 IDEF0 for smart manufacturing system. 36
Figure 21. Detailed IDEF0 for the smart manufacturing case study. 38
Figure 22. IDEF0 for data classification part. 40
Figure 23. Convolution neural network (CNN) layers (Jermyjordan 2019). 41
Figure 24. IDEF0 for the healthcare section (gene biomarkers). 42
Figure 25. The optimal hyperplane for binary classification by SVM (Liu et al. 2018). 45
Figure 26. Benefits of big-data analytics techniques (NIST, 2018). 47
Figure 27. iFactory at IMS lab, University of Windsor. 50
Figure 28 History of the four industrial revolutions. 52
Figure 29. Upgrading existing manufacturing systems (EigenInovations 2019). 54
Figure 30. Real-time /Non-real time data analytics (Steve Chadwick 2016). 55
Figure 31. Real-time monitoring using data analytics (Steve Chadwick 2016). 55
Figure 32. Normal class, good image (left), and defected class, bad image (right). 56
Figure 33. Structure of convolutional neural networks (CNN) (FreeCodeCamp, 2019). 57
Figure 34. Input image with its different feature maps. 58

Figure 35. Rectified Linear Unit RELU.....59
Figure 36. Softmax activation function.....59
Figure 37. Results based on the accuracy, batch size, and the number of epochs with dropout 0.1.....61
Figure 38. Results based on the accuracy, batch size, and the # of epochs with dropout 0.2.....62
Figure 39. Patient class distribution.....67
Figure 40. Multi-Class classification model with performance measures.....71
Figure 41. Node Four DS vs. Rest with six genes relations matrix.....73
Figure 42. Circos plot for the biomarker genes in node number two for the DR samples based on the correlation coefficient among genes expressions ($p<0.05$).....75
Figure 43. Circos plot for the biomarker genes in node number two for the Rest samples based on the correlation coefficient among genes expressions ($p<0.05$).....75
Figure 44. Circos plot for the biomarker genes in node number three for the LH samples based on the correlation coefficient among genes expressions ($p<0.05$).....76
Figure 45. Circos plot for the biomarker genes in node number three for the Rest samples based on the correlation coefficient among genes expressions ($p<0.05$).....76
Figure 46. Boxplots for the nine biomarker genes in node number three show the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (LH vs Rest).....77
Figure 47. Boxplots for the 10 biomarker genes in node number one show the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (DH vs Rest).....78
Figure 48. Boxplots for the 14 biomarker genes in node number two show the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (DR vs Rest).....79
Figure 49. The distribution of breast cancer subtypes samples in each class.....83
Figure 50. The distribution of breast cancer subtypes samples in each treatment therapies samples...84
Figure 51. Percentages of patient class distribution.....85
Figure 52. Schematic representation of the proposed models based on the linkage type.....86
Figure 53. The five linkage types: Single, Complete, Average, Centroid, and Ward's linkage.....88
Figure 54. Ward's linkage model: classification model with performance measures.....92
Figure 55. Ward's Linkage model DR vs. (DS, LS) Node with five genes relations matrix.....94
Figure 56. Boxplot for the biomarker genes in Ward's linkage model shows the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (DH vs. LH) and (DR vs. (DS, LS)).....95
Figure 57. Circos plot for the biomarker genes in Ward's linkage model for the DS class samples based on the correlation coefficient among genes expressions ($p<0.05$).....96
Figure 58. Circos plot for the biomarker genes in Ward's linkage model for the LS class samples based on the correlation coefficient among genes expressions ($p<0.05$).....96
Figure 59. Network genes pathway that includes most frequently altered neighbour genes for (DSCAM, MARK2, ROBO1).....98
Figure 60. Single linkage model: classification model with performance measures.....101

Figure 61. Complete linkage model: classification model with performance measures. 102
Figure 62. Average linkage model: classification model with performance measures. 103
Figure 63. Centroid linkage model: The classification model with performance measures. 104

LIST OF TABLES

Table 1. A sample of the literature review. 9
Table 2. Features comparison of DML, FMS, and RMS (Koren & Shpitalni, 2010). 24
Table 3. Manufacturing systems paradigms (H. A. ElMaraghy, 2005). 26
Table 4. Results (Accuracy and loss) With different batch sizes, Epochs, and Drop out. 60
Table 5. list of classes with the number of samples in each class, with the number of genes for each class after filter feature selections. 66
Table 6. illustrates the results of using mRMD 2.0 vs mRMR on each node then applying random forest classifier on each node. 69
Table 7. Gene biomarkers for each class versus the rest at each node. 70
Table 8. Comparison of the standard classifiers at each node of the proposed model. 72
Table 9. Class list with the number of samples in each class. 83
Table 10. Computing the distance between each pair of classes; di,j is the distance between classes i and j 87
Table 11. Ward ‘s linkage model: 37 biomarker genes. 93
Table 12. Single linkage model: 41 biomarker genes. 99
Table 13. Complete linkage model: 31 biomarker genes. 99
Table 14. Average linkage model: 34 biomarker genes. 100
Table 15. Centroid linkage model: 49 biomarker genes. 100
Table 16. A sample of the patient's clinical data 122
Table 17. A sample of the patient’s genes expression with desired classes 123

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ANN	Artificial Neural Network
BC	Breast Cancer
BD	Big Data
BDA	Big Data Analytics
BOL	Beginning Of Life
CBDM	Cloud-Based Design and Manufacturing
CC	Cloud Computing
CNC	Computer Numerically Controlled
CNN	Convolutional Neural Networks
CPS	Cyber-Physical Systems
DH	Deceased and Hormone
DL	Deep Learning
DML	Dedicated Manufacturing Lines
DPI	Dot Per Inch
DR	Deceased and Radio
DS	Deceased and Surgery
DSS	Decision Support System
EOL	End Of Life
FMS	Flexible Manufacturing Systems
GD	Gradient Descent
HaaS	Hardware As A Service
HDFS	Hadoop Distributed File System

I4.0	Industry 4.0
IDEF0	Function modelling methodology which refers to “ICAM DEFinition for Function Modeling" where ICAM is an acronym for "Integrated Computer Aided Manufacturing.”
IG	Information Gain
IMS	Intelligent Manufacturing Systems
IoT	Internet of Things
IP	Internet Protocol
KDD	Knowledge Discovery in Databases
KNN	K Nearest Neighbor
LH	Living and Hormone
LR	Living and Radio
LS	Living and Surgery
MAS	Multi-Agent Systems
ML	Machine Learning
MOL	Middle Of Life
MRI	Magnetic Resonance Imaging
mRMD	Maximum Relevance Maximum Distance
mRMR	Minimum Redundancy Maximum Relevance
NIST	National Institute of Standards and Technology
NN	Neural Networks
NSF	National Science Foundation
PLM	Product Lifecycle Management
RDBMS	Relational Database Management Systems
ReLU	Rectified Linear Unit

RMS	Reconfigurable Manufacturing Systems
RNN	Recurrent Neural Networks
SGD	Stochastical Gradient Descent
SMOTE	Synthetic Minority Over-Sampling Technique
SVM	Support Vector Machine
WCSM	World-Class Sustainable Manufacturing
XaaS	Everything As A Service

CHAPTER 1. INTRODUCTION

1.1 Overview

Data is a simple isolated fact that is not significant alone because it does not relate to other data and comes as raw observations, statistics, and measurements. These facts (data) are put, related, combined, and connected in a context, and information appears after the understanding of these relations. This information will help to identify patterns. By processing, investigating, and studying these patterns, we acquire knowledge. With a complete understanding of the outcomes and effects of such knowledge, future consequences can be predicted, decisions can be made, and choosing between alternatives become more comfortable, which will all lead to insight (ElMaraghy 2009).

Therefore, based on these definitions, it is clear that data is the primary source of information, which, in turn, is the primary source of knowledge, which is the primary source of insight. This hierarchy is shown in Figure 1 below. It is clear that the main reason for having the data, information, and knowledge is to have the ability to make the correct decisions.

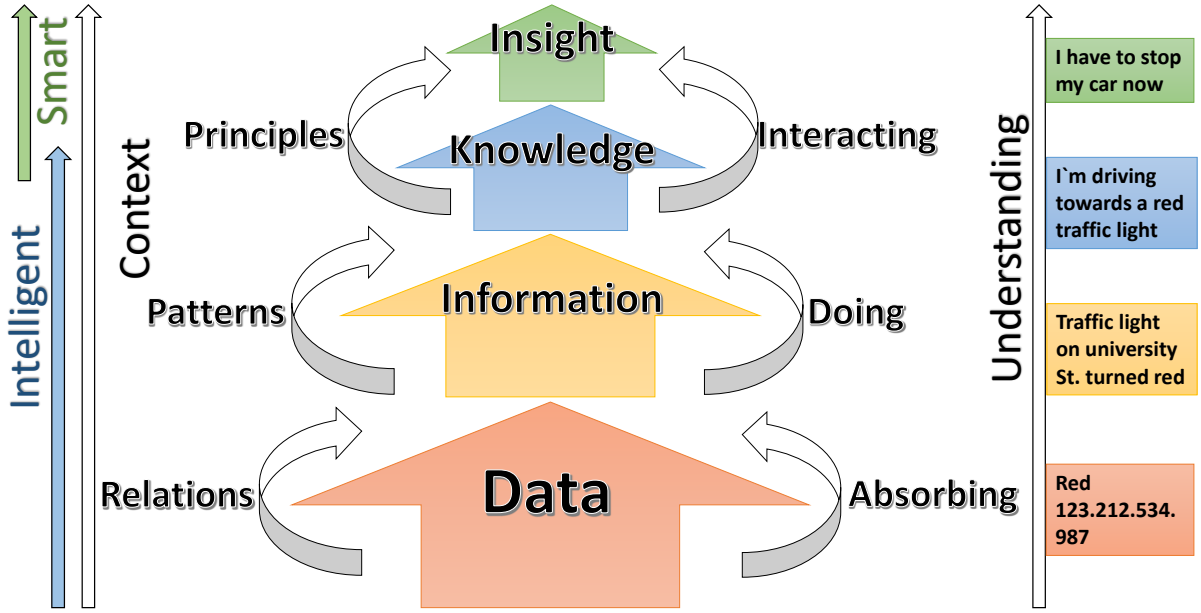

Figure 1. Data, information, knowledge, and insight.

What makes data big data? Big data is data that is too large to handle within a system, the data generation is too fast, and the data is in many different formats or types. Therefore, big data is characterized as large data sets with multiple dimensions, regardless of the source of information. These dimensions are called the big-data three Vs, which are volume, velocity, and variety. Volume means that the data is generated in massive amounts, velocity means the data is continuously generated (very quickly), and

variety means the data exists in different formats such as structured, unstructured, or semi-structured (Dumbill 2013).

According to the extensive use of big data in many domains, researchers identified some other dimensions other than the three Vs. These new dimensions are value, validation, verification, veracity, vision, and volatility. From the engineering point of view, these new dimensions (the new Vs) provide a better means of characterization for the data collected from and about manufacturing applications and processes (Berman 2013), (Zheng et al. 2013).

IBM introduced one of the proposed big-data dimensions shown in Figure 2 because it extracts business value as the fifth V from the four primary Vs, which are velocity, variety, volume, and veracity (Steve Chadwick 2016).

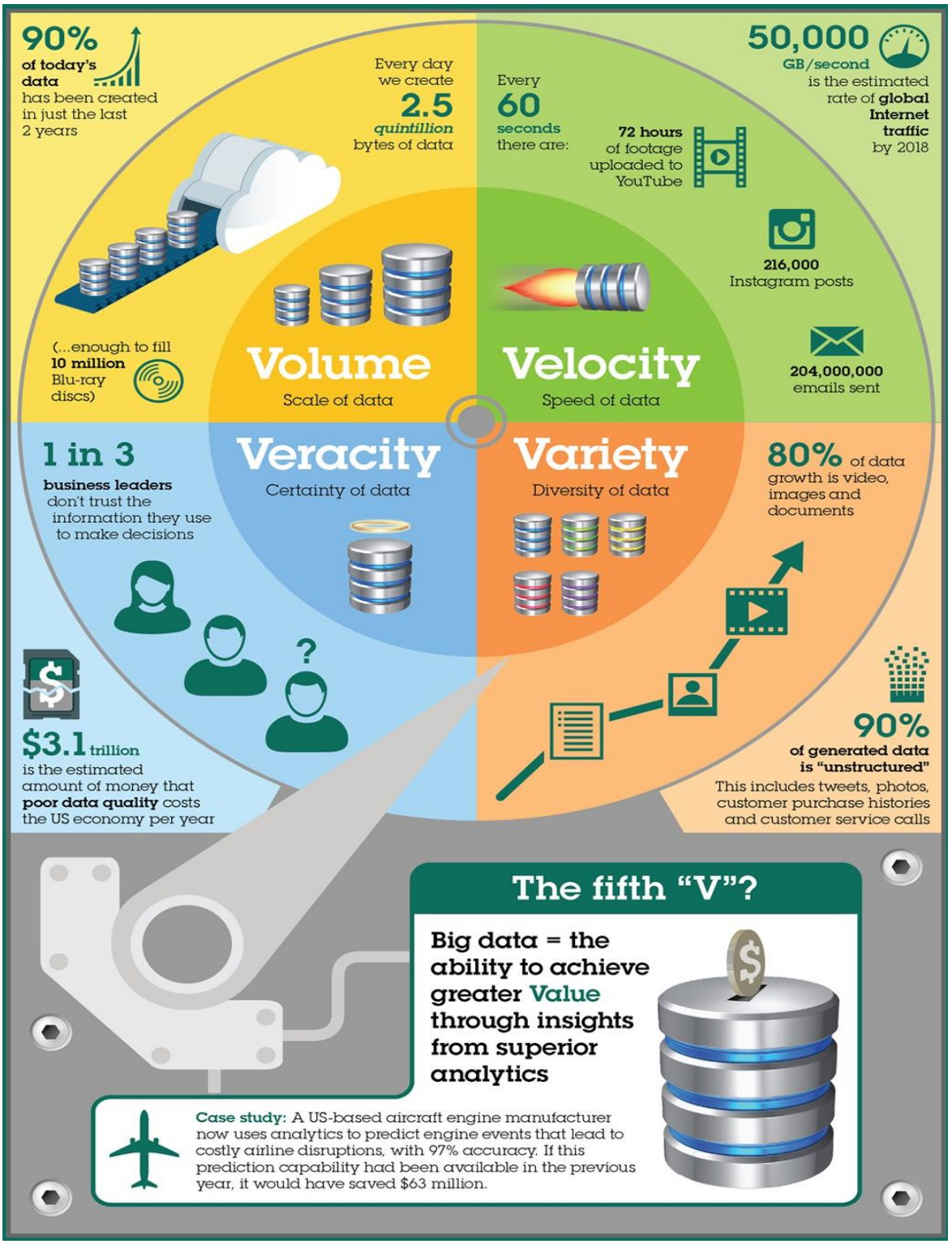

Figure 2. Extracting business value (fifth V) from the four main V's of big data by IBM (Steve Chadwick 2016).

1.2 Research Motivation

Although big data is still data and the foundations are the same, big and small data analysis necessitate the use of the same disciplines: mathematical statistics, probability theory, computer science, and visualization. While small datasets can be easily managed by traditional databases, which are generally

known as relational database management systems (RDBMS), big data cannot be analyzed using those traditional methods. As big data requires special means of storage and analysis because it is generated in a different format from small data and contains a lot of noise and redundancy in addition to useful information. Big data requires preprocessing (which includes filtering and indexing) before and after storing it in a big-data warehouse for analysis (Santos et al. 2017). The big-data storage phase must be involving the use of the fastest and most reliable computer network protocols to access and analyze the data easily. The storage mechanism has to be dynamic to adopt modern and dramatically-changeable networking technology. The streamlines of data have to be stored in modern storage servers with parallel processors for multitasking purposes. Because the amount of data is huge, the most efficient algorithms have to be utilized to store and process the data to produce reports for observers and decision makers quickly. Many statistical and computer science approaches are used to analyze the big data in all fields. However, there is a difference between managing big data in different domains, such as the manufacturing and healthcare fields and the management of big data in other fields because, in manufacturing, fast decision-making leads to not only cost saving but also high levels of sustainability and safety, which is a necessity. Hence, the most reliable and efficient algorithm has to be selected for collecting, preprocessing, storing, and analyzing data and making informed decisions in this area. Reducing processing time is the biggest challenge in this field, such as the time it takes for driverless, autonomous cars to make decisions. Mistakes can cost lives.

Therefore, this research is motivated by big-data analytics, which is the main enabler of many complex systems, such as smart manufacturing systems (the fourth industrial revolution), health informatics, and many others. The fourth industrial revolution is known as smart manufacturing (Industry 4.0). The idea behind Industry 4.0 is that it is different from the first three industrial revolutions, which were the results of advancements in mechanization, electricity, and IT, respectively. Now, the introduction of the internet of things (IoT) and cyber-physical systems (CPSs) to the manufacturing environment is leading to a fourth industrial revolution. In Industry 4.0, field devices, machines, production modules, and products are comprised of CPSs that autonomously exchange data and information, trigger actions, and control each other independently.

Moreover, one essential enabler of sustainable manufacturing in smart manufacturing (Industry 4.0), which is the scope of this research, is reliable systems for storing, analyzing, and synthesizing very large amounts of data (big data up to the scale of terabytes) throughout the whole manufacturing system lifecycle.

1.3 Problem Statement

Today's complex systems require rapid decision-making for improving system quality, productivity, and sustainability based on big-data analytics. However, different industries have different needs and different environments, as well as various types of data, to be utilized for decision-making. In general,

there cannot be one approach that solves all manufacturing decision-making problems. Some complex systems are not yet ready to manage and take full advantage of the big data that is available to them due to the lack of pipelined data acquisition, processing, analysis, and finally, decision-making procedures that are tailored to the specific manufacturing problem.

Therefore, it is necessary to adopt smart decision-making tools for big data that are tailored to the specific manufacturing problem of how to be responsive and adaptive in a dynamic market and continuous technology development. To achieve this purpose, there is a need to integrate cutting-edge technologies, such as big data analytics, machine-learning tools, and the IoT, to accomplish a pipelined process that can be used to solve the decision-making issues that are faced in modern complex systems.

Each system has it is own parameters and characteristics. Such pipeline processes must be able to identify the key characteristics in the system (e.g., the genes related to breast cancer or the vision features of the production line in a manufacturing system) to feed into the learning process. This may require initial human intervention to guide the identification and selection of features that are obtainable and relevant to the prediction problem at hand. Often, the various industries (whether it be manufacturing, healthcare, or otherwise) that desire to take advantage of big data for prediction and quality improvement have no clear idea of the characteristics that might be useful for their desired goal. This phase of handpicking relevant variables, and sometimes even creating new variables that are relevant to the goal of prediction, involve human-guided data mining.

It is essential for modern complex systems to tune their parameters based on the learning process. The development of a continuous learning process that can detect unusual events using the key characteristics of a system and predict outcomes and, hence, make a decision is the goal for our proposed systems. The proposed modern complex system is required to strive to improve performance and report any decline in performance measurements, such as the error, specificity, and sensitivity of the outcomes.

Therefore, this work claim is that, currently, whether it is in the healthcare or manufacturing industry, there is a lack of pipelined and problem-specific processes that take full advantage of big data to achieve improved outcomes.

1.4 Research Objective and Scope

Big-data analytics is a new research area and a key enabler for unlimited domains, including manufacturing, marketing, healthcare, information science, and business processes. For example, some of the important applications of big-data analytics in the different domains are shown in *Figure 3* below.

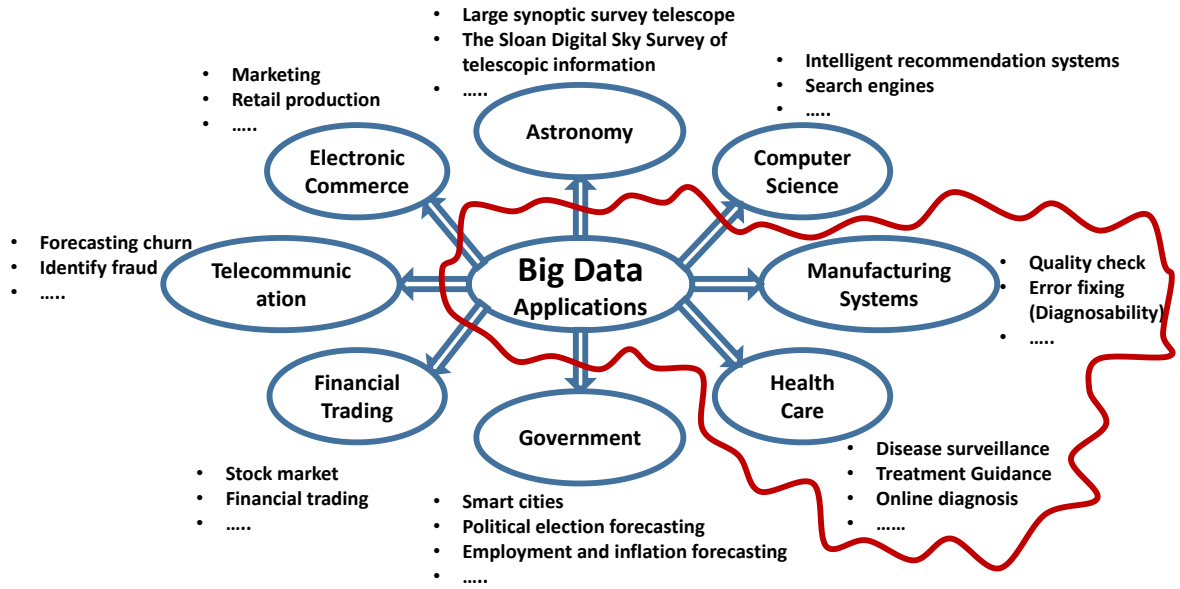

Figure 3. Significant applications of big data.

Big-data analytics is still not widely used in some domains, such as the manufacturing domain, in which the IoT and cloud computing should be integrated into the decision-making process (see the literature survey). These pieces of new technologies can potentially be exploited in modern systems. The essential big-data research is currently concentrated on capturing, processing, and analyzing big data (Li et al. 2017).

The literature has many proposed big-data analytics solutions for complex systems. However, there is a need to come up with comprehensive approaches that can put the different data analytic pieces together and utilize them in an integrative way. A modern complex system can integrate any new learning model and interact with other learning models with minimum human interference.

Nevertheless, the nature of the target outcomes to be improved via big-data analytics are various as well as the types of potential characteristics that are used as inputs, relevant to the outcomes. This makes it difficult to have one single procedure that fits and solves each and every problem in each and every domain. Intelligent outcome-improving decision-making procedures need to be tailored for each and every class of similar industry problems.

Therefore, our main objective in this thesis is to illustrate that such problem-specific data analytics solutions can be achieved in specific cases by carefully building a pipeline in which essential characteristics are identified, processed, fed into an appropriate data analytics modelling device, and finally used in making decisions about the target outcome. As shown in the red region of Figure 3 above, the scope and objective of this research are concentrated on the two domains, which are manufacturing systems and health informatics. To achieve our goal, we will use three case studies: two in healthcare settings and one in the manufacturing industry.

In summary, the objectives for the manufacturing domain are:

- Develop a new method or tool for gathering, analyzing, and processing big data (structured or non-structured) generated from manufacturing system facilities that support real-time decision-making and knowledge discovery using big data and data mining tools.
- Develop a tool that will support decision makers in designing and operating their manufacturing systems sustainably as well as economically.
- Build an integrative model for complex modern systems, which consists of big data, IoT, CPS, ML and DL. These components are represented in state of the art model for a real-time decision-making process.
- Add the missing enablers which iFactory lacks to convert it to Industrial 4.0 (Transfer I3.0 to I4.0).
- Extract hidden features in the product, which can predict the unusual situations (e.g. defective vs normal) product to speeds up the prediction performance.

While the objectives for the health informatics domain are:

- Develop machine learning models to find gene biomarkers, that are related to breast cancer survivability based on specific treatments.
- Develop a hierarchical one-versus-rest model for predicting five-year survival based on the treatment and gene expression.
- Develop a Semi-supervised learning model to find borders between classes groups in a multi-class learning problem.
- Develop a new method or tool for gathering, analyzing and processing big data (structured or non-structured) generated from a healthcare system.
- Develop a model that supports real-time decision-making and knowledge discovery using big data analytics and data mining tools.
- Techniques to overcome Class-Imbalance (C-I) were presented to handle what is known as the curse of dimensionality problem, where the number of samples is much smaller than the number of features.

1.5 Research Gaps and Novelty

From the literature review, it is evident that most researchers consider many things, but not a big-data analytics completely-pipelined process with methods and tools in some domains, such as manufacturing systems. In December 2016 at the IEEE International Conference on Big Data in Washington D.C. The National Institute of Standards and Technology (NIST) presented a summary of the Symposium on Data Analytics for Advanced Manufacturing with the theme of "From Sensing to Decision-Making," which highlighted the key challenges, issues, and needs for implementing big-data analytics for smart manufacturing systems (Narayanan et al. 2017). Then in 2018, the NIST reported that even when manufacturers have some data analytics tools, which are implemented with technical barriers that prevent the widespread adoption of data analysis in practice, they are not using the scale of big-data nor a real-time decision-making, which has led to limited success because there are delays in the decision-making process (NIST 2018).

Babiceanu proposed future work that focuses on big-data algorithms developed for manufacturing operations (Babiceanu et al. 2016). Also (Qiu et al. 2015) considered the benefits of sharing assets and services by having a system for these assets, an information infrastructure, and finally, a decision support system (DSS) for big data in manufacturing supply chains.

Some other researchers only discuss big-data requirements (Gölzer et al. 2015), while (Dutta et al. 2015) only presented a framework for implementing big-data projects in manufacturing systems that were validated by examining real implementations in Indian manufacturing companies. A new paradigm, namely cloud-based design and manufacturing (CBDM), was presented by (Wu et al. 2015) which is also not dealing with big data analysis. (Kashkoush et al. 2015) Introduced a knowledge-based model for the optimum assembly sequence for a product family. It was a data managing method that was applied for a manageable amount of data, not big data. Kumaraguru and Morris proposed continuous performance management with real-time analytics for manufacturing systems, but not for big data (Kumaraguru et al. 2014). A sample of the literature review is presented in Table 1 below.

Table 1. A sample of the literature review.

Paper	Manufacturing System	CPS	IoT	Cloud Computing	Big Data	Decision Making
National Institute of Standards and Technology (2018)	x	x	x	x		x
Babiceanu, R. F., & Seker, R. (2016)	x	x	x	x	x	
Baheti, R., & Gill, H. (2011)	x	x	x	x	x	
Gölzer, P., Cato, P., Amberg, (2015)	x	x	x	x	x	
Wang, L., Törngren, M., Onori, (2015)	x	x	x	x	x	
Borgia, E. (2014)	x	x	x	x	x	
Qiu, X., Luo, H., Xu, G., Zhong, R., (2015)		x	x	x		x
Xu, X. (2012)			x	x		
Dutta, D., & Bose, I. (2015)	x				x	
Kumaraguru & Morris, (2014)	x	x	x			
Wu, D., Rosen, D. W., Wang, L., (2015)	x	x	x	x	x	

For a long time, companies used to get rid of data because they had too much data to manage. However, with the existence of big-data tools like the Spark ecosystem, they can now manage massive amounts and different types of data. The stored data, with its different formats and huge size, complicates processing and analysis. Therefore, some machine-learning tools are used to create metadata catalogues that can help data analysts to develop decision-making models. Therefore, companies tend to use big data only for tracking purposes, not as a basis for improving operations.

From the wide and critical literature review, we can identify the research gaps in big data in general, and especially in manufacturing systems and healthcare domains. As for the big-data gaps, researchers

indicated that they most likely result from data variety, not data volume, because heterogeneous data mixing is more significant than the data volume. Therefore, the emphasis is not on the "big" part of the big data but, rather, the problem of data variety. In these two domains, research gaps are identified as not only the lack of decision-making methods and tools for big data but also how to manage this huge sort of data, starting with collecting, storing, analyzing, securing, and processing this data — taking the variety, velocity, and volume of data into considerations. Finally, integration solutions needed to connect all the data sources (software and hardware technologies) to create data standers that are manageable (Obitko et al. 2013).

It is clear that the industrial automation and manufacturing sector is already a large producer and consumer of a lot of data. During the last few decades, manufacturing companies demonstrate significant productivity gains regarding both improved quality and efficiency based on the utilization of processed data and advanced methods of data analysis. Today, the big-data paradigm is a promising concept for another substantial wave of gains to achieve improved efficiency in design, production, product quality, and achieving customers' needs. According to (Manyika et al. 2011), big data can help manufacturers to reduce their product development times by 20% to 50%, as well as defects before production through simulation and testing. By using real-time data, companies can also manage demand planning across extended enterprises and global supply chains while reducing defects and rework within production plants (Obitko et al. 2013).

From the above literature, we can claim that the significant gaps in the literature on big-data analytics in the various industries is that there seems to be no published research about the employment of a complete pipeline to take full advantage of big data (concerning volume, variety, and velocity) in real-time decision-making in the manufacturing industry. And also the claim that there could exist a single big-data analytic pipeline that can solve every problem in all domains. Such claims have been made without substantiating them. Our aim in this thesis is to fill this understanding gap and provide a few case studies that support the need for tailoring different data analytic pipelines for different situations and that such pipelines need to be partially human-guided. Nevertheless, some generality can be achieved within classes of similar issues. In particular, in this thesis, we consider case studies in which the desired decision-making problems are based on the classification of the outcomes of interest.

1.6 Thesis Hypothesis

Human-guided big-data analytic pipelines can be used as decision-making tools for specific categories of similar issues that are relevant to the manufacturing and healthcare industries.

We use three case studies to verify this hypothesis: two in healthcare and one in manufacturing. Throughout our proposed pipeline/solution, we emphasize how it is essential to tailor the big-data analytic loop to the specific characteristics of the issues at hand and that there cannot be a single pipeline that can fit all types of decision-making problems. Nevertheless, we propose some pipeline algorithms

that extend from data acquisition and feature selection (both human-guided and automatic) to the prediction of the desired outcomes in real-time, which have some degree of generality within the classes of industry issues that are considered here. These pipelines are designed to accommodate large datasets and use cloud-based computational tools to build and select models and use new input data for predicting the desired outcomes.

For instance, in the manufacturing industry, the proposed decision-making pipelines are expected to enhance the overall outcomes of the production system, including the quality of the products, performance measurements, as well as reduce the cost by speeding up the decision-making process.

CHAPTER 2. LITERATURE REVIEW

2.1 Overview

In this section, I discuss the main components of big data analytics for complex systems, such as the smart manufacturing complex system (Industry 4.0). A definition and extensive literature review for each component are represented in this chapter.

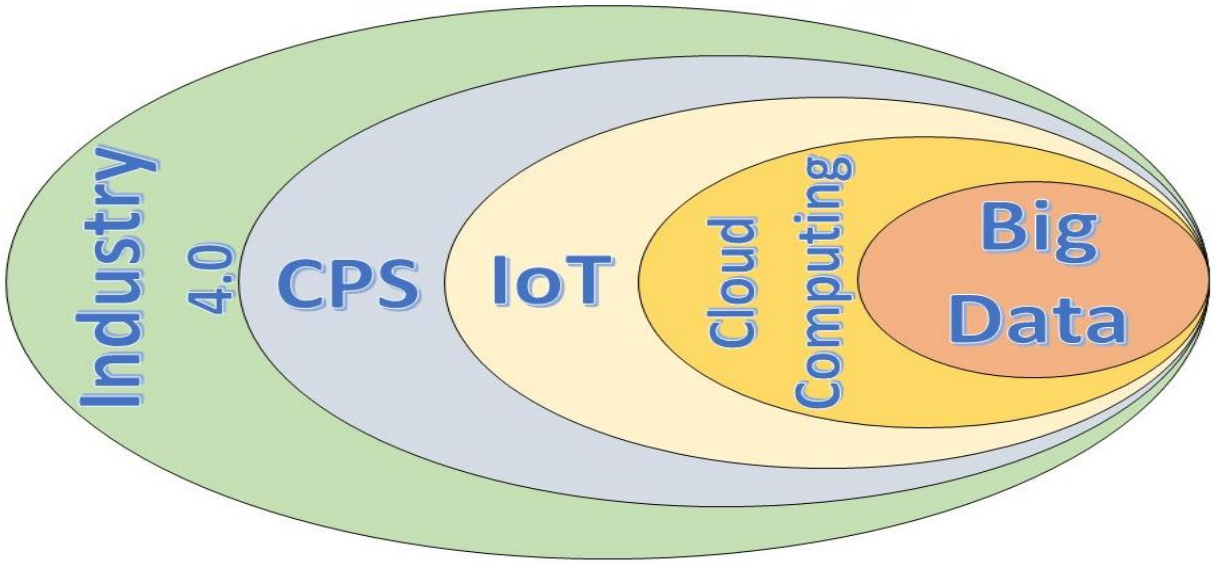

Figure 4. Components of smart manufacturing (Industry 4.0).

2.2 Big Data Analytics in Complex Systems-Related Domains

The implementation of project phases in big data in manufacturing domains are somehow similar to other big data implementation projects in other domains. As presented by (Dutta et al. 2015), when they successfully implemented a big data project at an Indian company (Ramco Cements Limited), they stated that any big data project implementation has three main phases and ten detailed steps. The three main phases are strategic groundwork, data analytics, and implementation, and the ten detailed steps are stating the business problem, research, team formation, creating a roadmap for the project, the examination and collection of data, data modelling, data analysis, IT system integration, and professionals training, which is shown in Figure 5. Therefore, manufacturing domain projects not only follow the same phases or steps but also use the already-available visions and understandings from related domains, such as logistics and supply chains (Dutta et al. 2015).

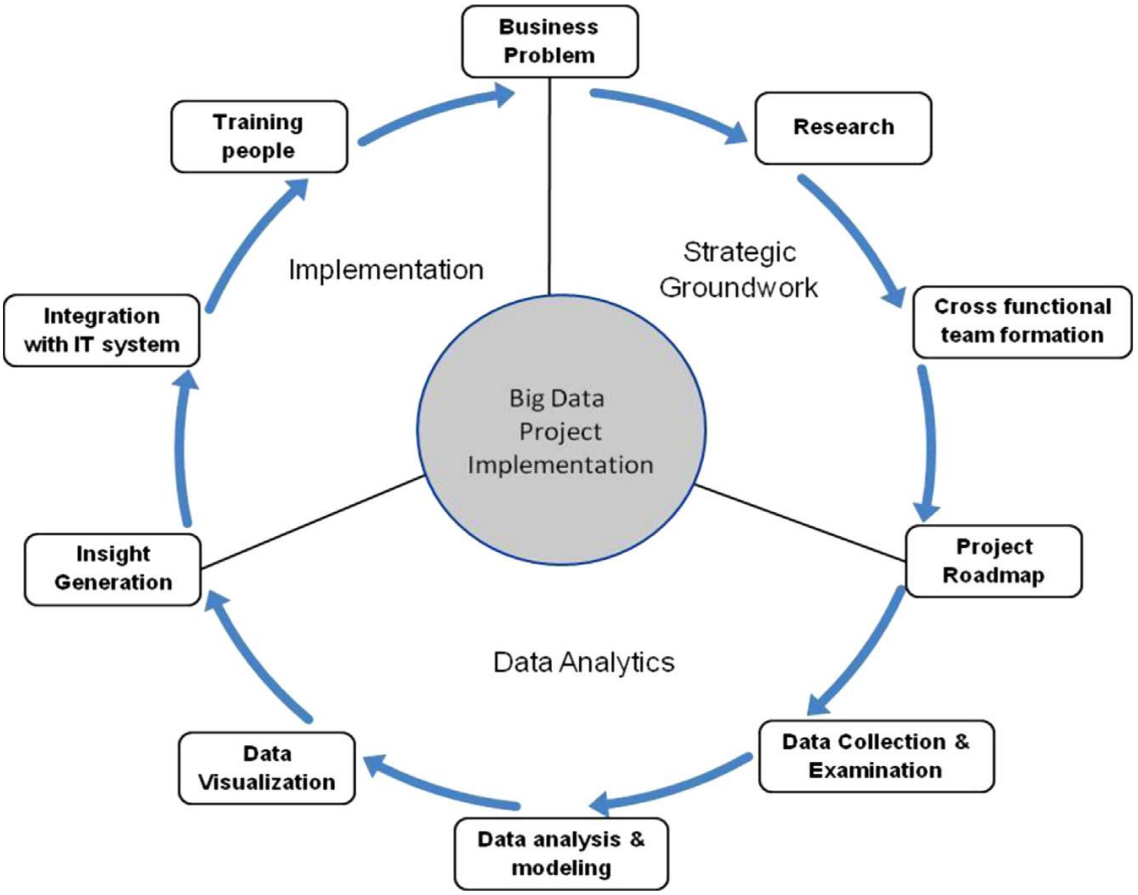

Figure 5. Big Data project implementation phases (Dutta & Bose, 2015).

McKinsey stated that by applying advanced analytics, which is the application of statistics and other mathematical tools to assess and increase performance, manufacturers could enhance their production processes by reducing waste and improving quality and yield (output per unit of input) by applying data analytics. This can be achieved by processing the historical data, identifying relationships and patterns between the process stages and inputs, and then improving the features that have the most significant effects on yield (see Figure 6). The most critical point how big is the generated data is because most companies only use big data for tracking purposes, not as a basis for improving operations (Auschitzky et al. 2014).

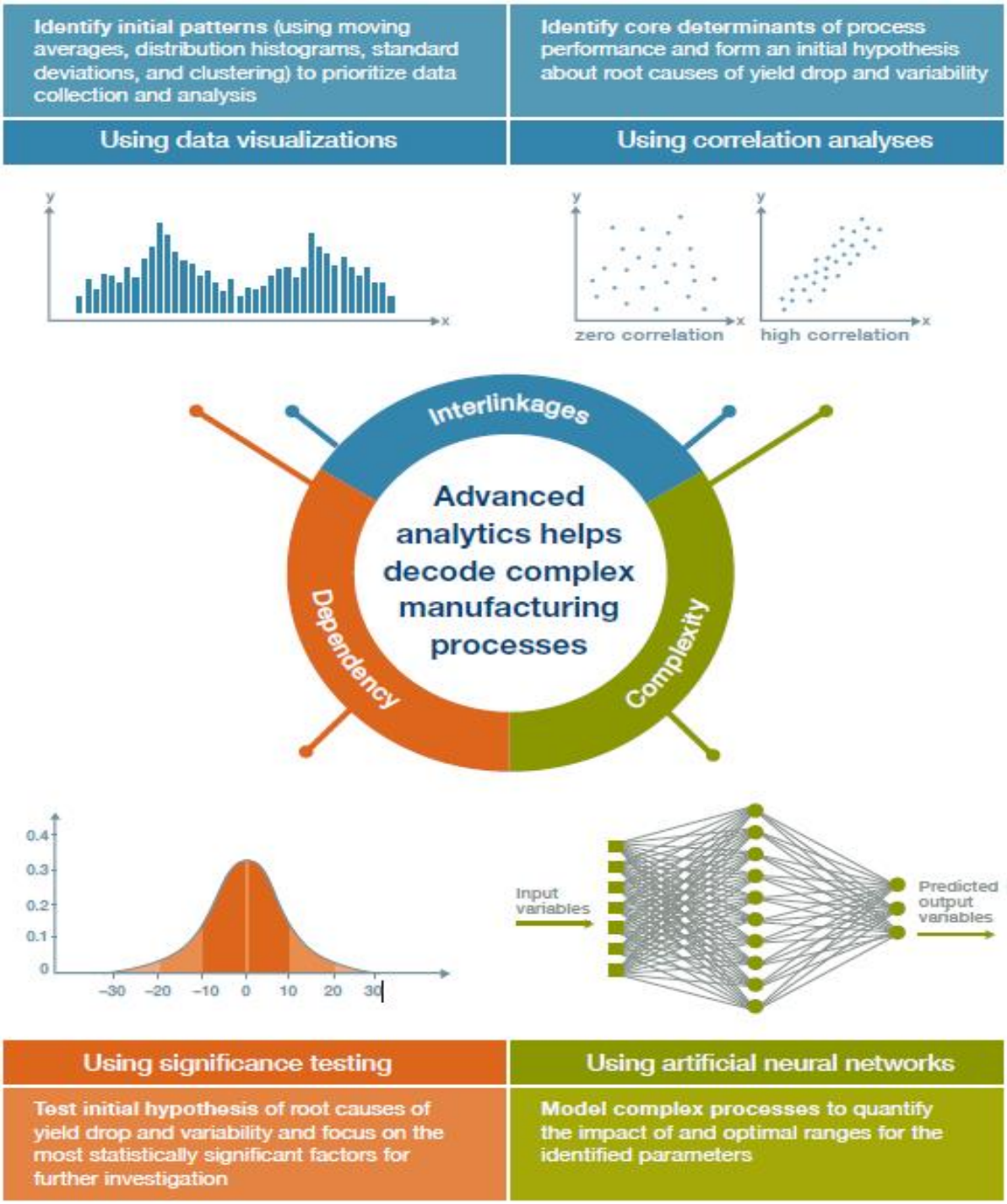

Figure 6. Advanced analytics for complex manufacturing processes (Auschitzky et al., 2014).

(Dubey et al. 2016) Proposed a theoretical, conceptual framework to investigate the importance of big data analytics (BDA) for world-class sustainable manufacturing (WCSM), which is a set of practices that lead to superior sustainability performance. While (Glawara et al. 2015) presented an approach for product quality management by breaking down the data from the production facilities into condition monitoring data, wear data, quality data, and production data and then linking them with data mining methods for deriving patterns, aiming to have zero-defect production. As is well known, conducting proper maintenance measures at the right time is the main reason for product quality, secure plant

availability, and process efficiency in manufacturing systems. (Hazen et al. 2014) Stated that data quality is a significant step in analytics and proposed a method for controlling and monitoring data quality in supply chain practices. Stating the problems and issues and suggestions for research.

(Li et al. 2015) stated that the vast amount of data related to the three main stages of product lifecycle management (PLM), which are beginning of life (BOL), middle of life (MOL), and end of life (EOL), are analyzed and concluded. The framework of big data in PLM is shown in Figure 7 below.

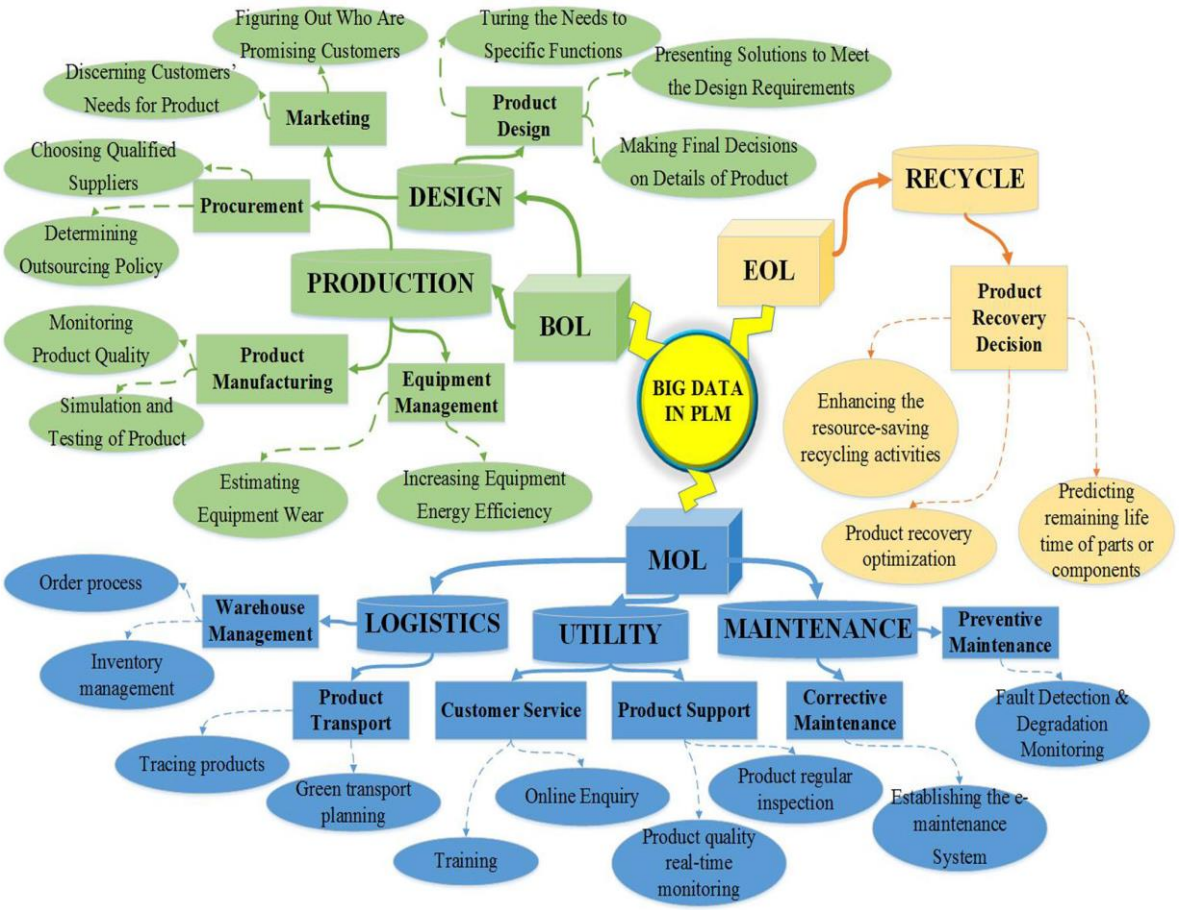

Figure 7. The framework of big data in PLM (J. Li et al., 2015).

(Obitko et al. 2013) Discussed some essential points about big data, by stating that the key output of BDA is to have enhanced data-driven decision-making. They also mentioned some related challenges like data search, data visualization, decision-making converted to action, and parallel big data processing. The multi-agent systems (MAS) can be employed for the last challenge. The MAS methodology is a very appropriate complement to big data processing. With its high-performance computing and parallel programming tasks, the spatial nature of big data be solved with MAS and the heterogeneity of data sources as well as the real-time or near real-time data processing because the MAS has some real-time processing benefits, like increasing efficiency and speed due to asynchronous and

parallel computation, increasing robustness (the systems degrade smoothly with agent failure), and increasing scalability (agents can be added when required).

(Zhong et al. 2015) proposed a significant big data methodology as a framework with its key stages starting with data collection, cleansing (to remove the noise), compression, classification, pattern recognition, machine learning, KDD, and decision-making.

Techniques and tools are required to process and analyze big data, such as log, text, audio, video, or other files. These tools have some key resources, such as memory, storage, network, and processing capabilities. For example, the most known one is the Apache Spark Ecosystem (Loshin 2013).

2.3 Data Mining, Artificial Intelligence, Machine Learning, and Neural Networks (Deep Learning)

The new generation of artificial intelligence (AI) simulates, extends, and stretches human intelligence. AI replaces the need for humans to analyze, optimize, judge, and make decisions through machine perception, machine learning (ML), machine thinking, and intelligent behaviour, such that AI ultimately provides intelligence for systems. Therefore, as the most disruptive technology in the world, a new generation of AI technology called AI 2.0, which includes ML, big data, cloud computing, the IoT, and other cutting-edge AI technologies, is profoundly influencing and changing manufacturing systems worldwide (Cheng et al. 2019). The relationships between different AI disciplines are shown in Figure 8.

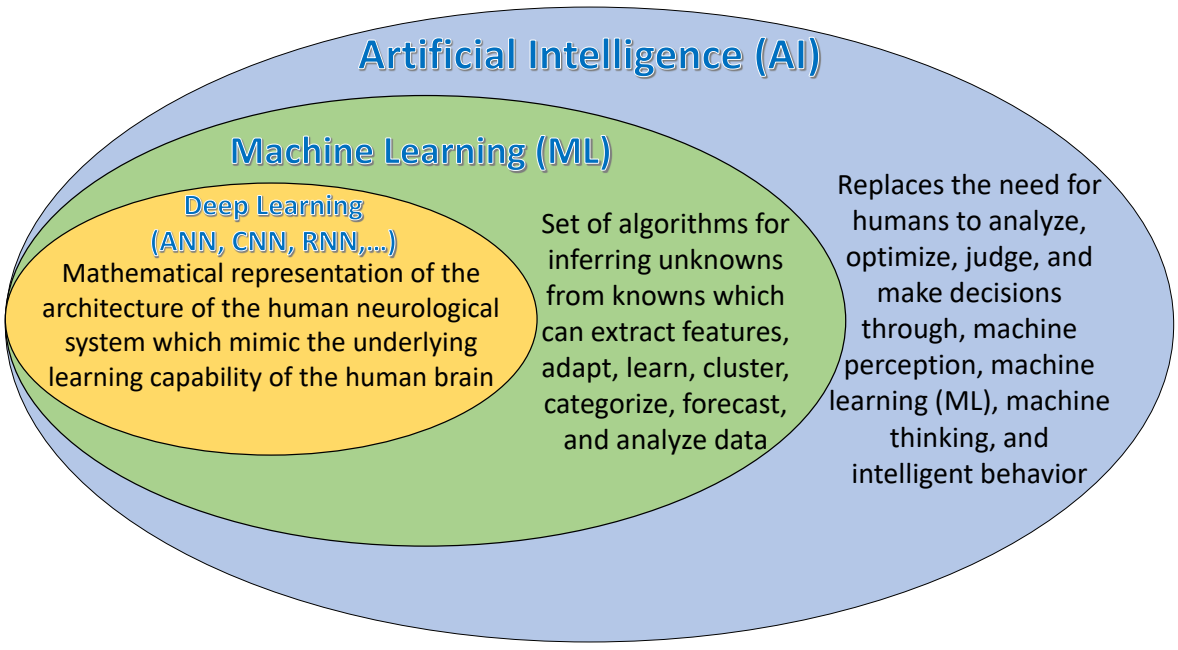

Figure 8. The relationship among different AI disciplines.

2.3.1 Data Mining

Data mining (DM) is an interdisciplinary task that is usually defined as the process of extracting valid, unknown, precise information from different huge databases to adjust and improve business decision-making (Fayyad et al. 1997). DM is based on computational intelligence, such as artificial neural networks, machines learning (association rules and decision trees), and advanced statistical techniques, such as logistical regression, that have created new intelligent tools for the extraction of useful knowledge and information. Nowadays, and due to a large amount of data generated by complex systems, traditional data analysis methods are no longer the best choice to use. DM approaches have created new intelligent tools for automatically extracting useful information and automatically. Therefore, the main objective of DM is knowledge discovery, and DM methodology is a technique used to extract predictive information and knowledge from databases (Wang 2007).

The main techniques of DM are classification, prediction, association, description, and clustering. These techniques involve applying some algorithms to extract hidden patterns, structures, associations, and differences from huge databases. In complex manufacturing systems, a vast amount of data that is generated about the product lifecycles, such as that about design, planning, quality check, maintenance, and fault finding, are stored in databases and data warehouses. DM is a significant tool for extracting knowledge from these data sources. The two main DM techniques that are most used in manufacturing are description and prediction. Predictive DM emphasizes forecasting model behaviour and predicting future values of variables based on current information from databases, while descriptive DM involves discovering remarkable patterns with which to describe data. Prediction and description goals can be accomplished using different DM techniques and tools (Choudhary et al. 2009).

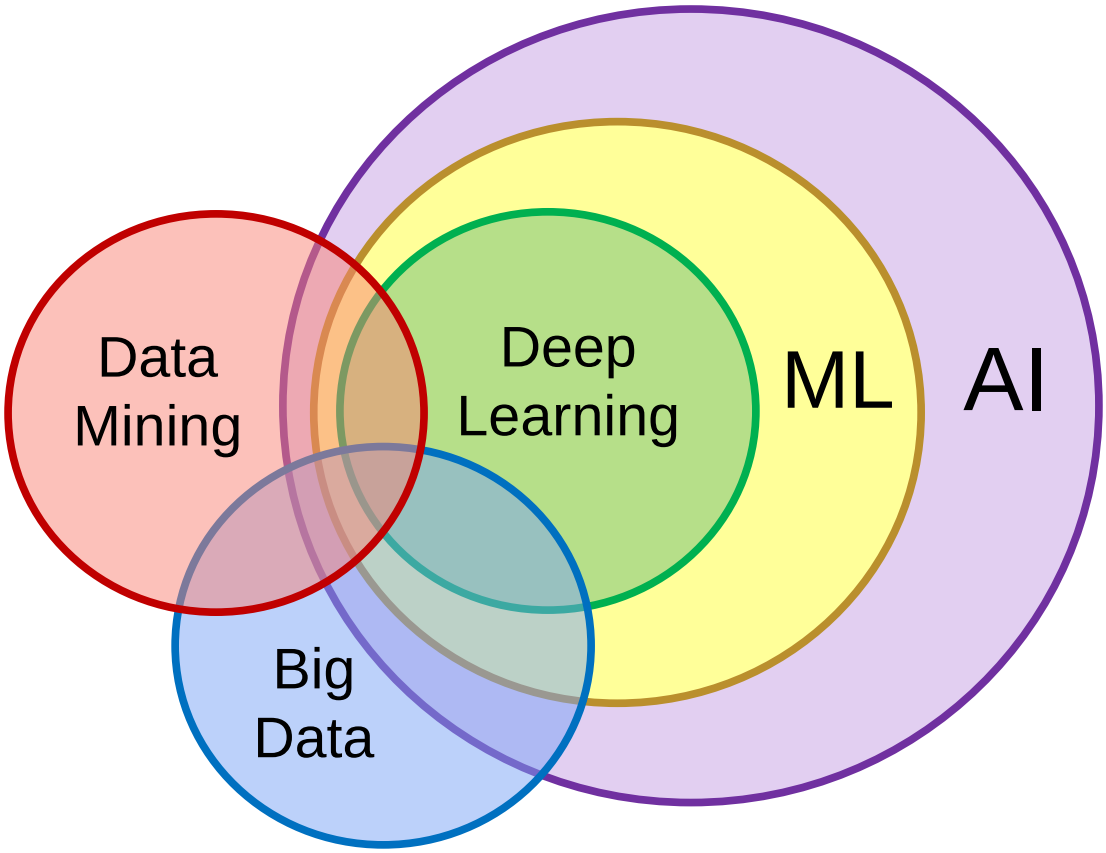

Figure 9. Venn diagram for AI, DM, and big data.

The Venn diagram shown in Figure 9 illustrates the overlapping of multidisciplinary AI, DM, and big data analysis.

2.3.2 Artificial Intelligence

In attempts to make machined have human functions, AI is considered to be an emerging interdisciplinary subject that involves theory, methodology, technology, and system applications that integrate cybernetics, statistics, computer science, mathematical logic, neurophysiology, and other disciplines and is used to mimic, extend, and stretch human intelligence. In industrial engineering, AI is used to make machines do several tasks and become highly human-independent. With that being mentioned, revolutionary AI reduces the time and cost of production and improves the quality of products, and machines can adapt to their surrounding environments, which means they can be aware of any unusual behaviour, decision-making, and most importantly, learn from previous experiences (JU et al. 2018, Zhang et al. 2018).

Data is the new gold, and extracting information from raw data can create revenue for any organization. From predicting customer's potential product interests to clustering customers for targeted sales packaging, all of these examples of the new dimensions of DM in different industries, and manufacturing systems are not exceptions.

The iFactory can generate large amount of data, including performance reports and logs. Machines sensors can also add to that amount of data. Making sense of the generated data requires collaboration from two fields, DM to extract information from the raw data, and big data to facilitate the access to data and store the new data. From this perspective, we are using DM as a tool to extract actionable information from (usually) very big data generated from iFactory. DM is considered to be part of ML.

2.3.3 Machine Learning (ML)

Machine Learning Definition (algorithms for inferring unknowns from knowns). ML represents a powerful set of algorithms that can extract features, adapt, learn, cluster, categorize, forecast, and analyze data, amplifying our understanding of things like obesity and our capacity to make predictions with unprecedented precision. ML application examples include Netflix, Alexa, speech recognition, and Tesla driverless cars.

A keyword in the field of ML is "finding patterns" in big data and using and learning from those patterns to make predictions. These predictions will be decisions or assist humans in making decisions. Therefore, the essence of ML is finding an objective function f that forms the best mapping between an input variable X and an output variable Y (i.e., $Y = f(X)$). The goal of ML is to find this optimal mapping to enable more predictions of outputs. The types of ML can be classified into three categories (Cheng et al. 2018) based on the differences among input samples:

2.3.3.1 Supervised Learning

If the learning process can be guided by prior knowledge, then this type of ML has supervised learning. The sample I has both an input feature vector xi and a corresponding output label, yi. Because we know label vector yi, a supervised algorithm can learn yi and categorize the new samples that have no label. If the variable yi has a continuous value, then the supervised learning approach involves the use of a regression model while if the value is categorization-based (e.g., "cancer," "normal" samples or "defected," "normal") products, then the supervision problem is called classification. As an example of supervised learning in regression (DeGregory et al. 2018), linear regression is the most commonly-used model for characterizing the relationship between a dependent variable and one or more explanatory variables (Bishop 2006). Logistical regression differs from linear regression by predicting categorical outcomes or classifying outcomes. In obesity, this is most relevant to the classification of obesity-related disease states or risks.

In this work, we have the supervised learning classification model showcased in Case Study 2, "A machine learning approach for identifying gene biomarkers guiding the treatment of breast cancer."

2.3.3.2 Unsupervised Learning

Sample i has only an input features vector xi; no corresponding output label yi exists. The goal of the learner is to group all samples in the sample space, using techniques such as cluster analysis.

In this proposal, we have an unsupervised classification model showcased in Case Study 3, "A novel approach for identifying relevant genes for breast cancer survivability on specific therapies."

2.3.3.3 Semi-Supervised Learning

Semi-supervised learning can also be classified into a single category similar to the above two categories of ML. Here, semi-supervised learning possesses features and partial labels, and it contains classification, regression, and clustering methods. Only one or two of them are included in supervised learning and unsupervised learning (Yu et al. 2018).

In this proposal, we have a semi-supervised classification model showcased in Case Study 3, "A novel approach for identifying relevant genes for breast cancer survivability on specific therapies."

2.3.3.4 Decision Tree Analysis

Decision tree learning is a predictive algorithm that uses both categorical and numerical data to assign samples to specific classes (Bishop 2006). Here, we will use decision trees to group subjects based on their characteristics into classes of varying health risks based on blood pressure and body fat percentage cut-offs. Unlike regression models, which rely on the minimization of error through least squares, decision tree analysis involves determining thresholds derived from input data. For example, falling above or below the threshold for patient age moves the patient into the appropriate class. While decision trees are effective standalone models for classification, the algorithm's performance is, in some cases, improved by using random forests, which aggregate the results of randomly-generated decision trees to produce an effective model (Bishop 2006).

2.4 Neural Networks

An artificial neural network (or simply, a neural network) is a mathematical representation of the architecture of the human neurological system and, hence, falls under the field of AI. Neural networks mimic the underlying learning capabilities of the human brain. The neural network is modelled such that a series of neurons (or nodes) are organized in layers in such a way that each neuron in one layer is connected to neurons of other layers with associated weights.

Neural networks are used to predict both continuous numerical and categorical data. There are two phases of effective neural network development. The first phase is what is known as training or learning. During the training phase, weights that are associated with connections between nodes are adjusted until the model performs well. This completed model is then applied to new data to make a prediction. This application of the model is called the testing phase. There are many variations of neural networks that are adapted for different conditions and applications (Bishop 2006).

2.4.1.1 Deep Learning

For many tasks, it is difficult to know what features should be extracted to feed to the AI algorithms. Aiming at learning feature hierarchies with features from higher levels of the hierarchy formed by the composition of lower level features (Bengio 2009), deep learning methods have the potential to overcome the deficiencies above in current intelligent fault diagnosis methods (Hinton et al. 2006).

In deep learning methods, automatically learning features at multiple levels of abstraction enable systems to learn complex functions. For machines to learn about these complicated functions, deep architectures are needed that are composed of numerous levels of non-linear operations. Through the deep architectures, deep learning-based methods can adaptively capture representative information from natural input signals through non-linear transformations and approximate complex non-linear functions with a small margin of error. Deep learning is part of ML. It was invented based on an artificial neural network (ANN) with multiple processing layers that tries to obtain hierarchical data representations. There are deferent deep learning techniques, such as CNN, recurrent neural networks (RNNs), and ANNs. The following is a brief about deep learning techniques (Zhao et al. 2016).

2.4.1.2 Convolutional Neural Network (CNN)

(LeCun et al. 1990) Proposed the use of CNN for image processing. It has two main properties: spatial weights and spatial pooling. It was sufficient for numerous computer vision applications (Krizhevsky et al. 2012) for which the input data usually has two dimensions. CNN was also successfully applied in different sequential data applications, such as natural language processing and speech recognition (Kim 2014). CNN is used to extract decisive features using two layers, convolutional and pooling layers. The convolution layer preserves the spatial associations among the drawn data (pixels) by learning image features using sliding windows of data input, which are relatively small, such as 7x7 or 9x9. Then these sliding windows are multiplied by the image matrix. The sliding windows are called kernel filters. The kernel filters slide from the left to right of the image with one or two overlapping pixels. Changing the values of kernel filters will produce different feature maps for the same input image. The output of this layer is the dot product between the kernel filter matrix and the concatenation matrix representation. This process is called a convolution operation. The second layer is the pooling layer. It aims to learn

the objects regardless of their places in the images and to decrease the length of the feature map. This reduces the number of the model's parameters.

2.4.1.3 Recurrent Neural Networks (RNNs)

(Schmidhuber 2015) Indicated that the RNN is one of the deepest neural networks. It can also address and create memories of sequences of input patterns. Connections between units from the directed cycle can also be built using RNNs.

Different from a primary neural network in which multi-layer perceptron can only map from input data to target vectors, an RNN can map from the entire history of previous inputs to target vectors in principle and allow the memory of prior inputs to be kept in the network's internal state. RNNs can be trained by backpropagation through time for supervised tasks with sequential input data and target outputs.

2.4.1.4 Artificial Neural Networks (ANNs)

ANNs are believed to be the most commonly-used algorithm (Hopfield 1982). In its most popular form, there are three components in an ANN: the input layer, a hidden layer, and the output layer. Units in the hidden layer are called hidden units because their values are not observed. ANN is an intelligence technique based on several simple processors or neurons. The circles labelled "+1" are intercept terms and are called bias units. Figure 10(a) is a human neuron, and Figure 10(b) is a simple model of an ANN. aij represents the jth neuron unit in the ith layer.

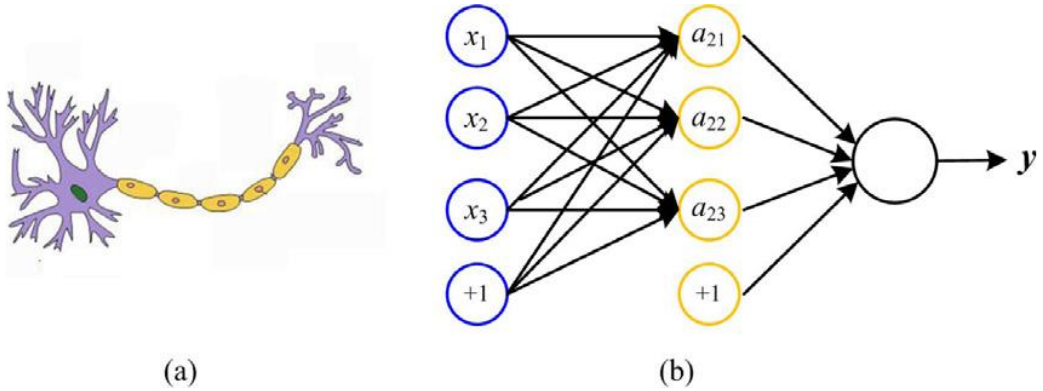

Figure 10. Human neuron and A multilayer perceptron (MLP) with two hidden layers (Liu et al. 2018).

The "neuron" in ANN is a computational unit that takes as input x1; x2; x3 and an intercept term. The output y can be obtained using the following formula:

$$y = f(W^T x) = f\sum_{i=1}^{3} W_i x_i + b \qquad (1)$$

Where f is the activation function, there are many activation functions. Sigmoid is the most common one and is usually used for ANNs. b is a scalar. The interconnections between many neurons are represented as a network of neurons in which the input of one is the output from the one in the previous

layer. This interconnection is what simulates and achieves human neurological performance. W is the weight matrix, where the values are updated during the learning process in an iterative procedure. W is updated during the forward-backward learning process in which the ANN computes the error and improves the learning rate based on a cost function (Liu et al. 2018). Many cost functions can be used. Gradient descent (GD) and Stochastic gradient descent (SGD) are commonly used to optimize cost reduction based on the learning rate (α). The vanishing problem in GD during deeper ANNs pushed for better optimization methods, such as dropout and ADAM (a method for stochastic optimization method (Kingma et al. 2014).

2.4.1.5 K Nearest Neighbour

K nearest neighbour (KNN) the idea of this classifier is simple, highly efficient and effective. It assigns a class to a sample point based on the smallest distance to a previously classified neighbouring point. In two-dimensional dataset in Figure 11. A simple Euclidean distance can be used to calculate the distance between the test data and the input data for class prediction.

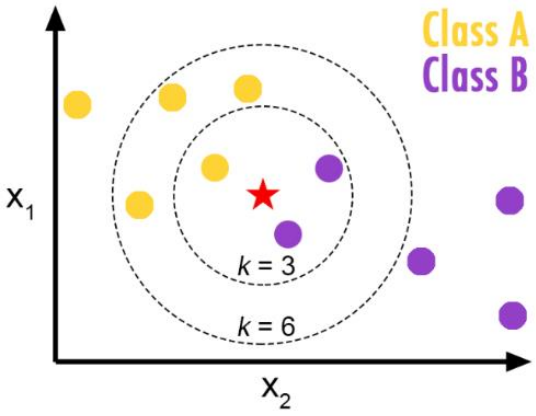

Figure 11. 2D data set displayed on a graph.

KNN was presented by (Fix et al. 1951). However, it is still widely used for ML applications, such as pattern recognition, text categorization, ranking models, and object recognition and event recognition applications. (Bhatia 2010) discussed many different variations of KNN that were derived from the basic model in detail. The basic and straightforward characteristics of this approach are one of its main advantages. The biggest drawback is that it is heavy on computation time because of the exhaustive distance calculation between every data point. However, because of the evolutional computational resources, it is still feasible to use KNN.

White noise is one of the most common problems encountered in data processing, especially image processing. In image processing, filters for noise reduction and eliminations are designed based on KNNs. NVIDIA implemented a filter based on KNN to remedy this issue (Kharlamov et al. 2007). Deng et al. focused on scaling a KNN for big data applications. The results showed that the proposed

KNN classification worked well in terms of accuracy and efficiency and, thus, its appropriateness for use in dealing with big data.

2.5 Manufacturing Systems Paradigms

Nowadays, manufacturing companies are facing frequent increasing and unpredictable market changes due to market competition and demand for new products, new product variants, and customized products. Therefore, to be competitive, manufacturers must have high-quality manufacturing systems and reasonable product prices but also the ability to quickly respond to market changes from the demand side, as well as from the variety perspective. Therefore, manufacturing systems should be immediately adaptable to market changes, product changes, government regulation changes, and system failures (Koren et al. 2010).

In this section, we identify three distinct manufacturing systems paradigms: dedicated manufacturing lines (DMLs), flexible manufacturing systems (FMSs), and reconfigurable manufacturing systems (RMSs). A comparison between the three paradigms is shown in Table 2 Below.

Table 2. Features comparison of DML, FMS, and RMS (Koren & Shpitalni, 2010).

	Dedicated	**FMS/CNC**	**RMS/RMT**
System structure	Fixed	Changeable	Changeable
Machine structure	Fixed	Fixed	Changeable
System focus	Part	Machine	Part family
Scalability	No	Yes	Yes
Flexibility	No	General	Customized (around a part family)
Simultaneously operating tools	Yes	No	Possible
Productivity	Very high	Low	High
Cost per part	Low (For a single part, when fully utilized)	Reasonable (Several parts simultaneously)	Medium (Parts at variable demand)

For the first paradigm, the DML has a fixed system and machine structure with very high productivity but neither scalability nor flexibility, and it produces specific products or parts at a large scale for a long time without having a wide product variety, which means we have mass production with limited product variants. Achieving this mass production will lead to cost savings only in the case of full capacity operation. Therefore, DMLs are sufficient in high-scale (mass production) and low-scope (fewer

varieties) production (i.e., whenever we need to have many varieties, using a DML will not be the right solution. Therefore, DMLs have high throughput and limited flexibility (Koren et al. 2010).

The FMS paradigm has a changeable system structure and a fixed machine structure. Therefore, it can be used to solve the flexibility issue of the DMLs by providing the ability to produce varieties of goods in reasonable volumes. This can be done through general-purpose computer numerically controlled (CNC) machines, but it increases the overall system cost. Therefore, FMSs solve the flexibility issue but create high equipment costs and do not have high throughput like DMLs.

On the contrary, the last manufacturing system paradigm is RMSs. It fills the gaps of both the previous paradigms by having the ability and flexibility of production when needed and as needed (ElMaraghy 2005). As shown in *Figure 12* below, DMLs (mass production) have a low scope and high scale (minimal varieties) while FMSs (mass customization) have a medium scope and medium scale. The Job Shop has a high scope and low scale (minimal quantity), and finally, RMSs can fit anywhere in the chart by adjusting scale and scope.

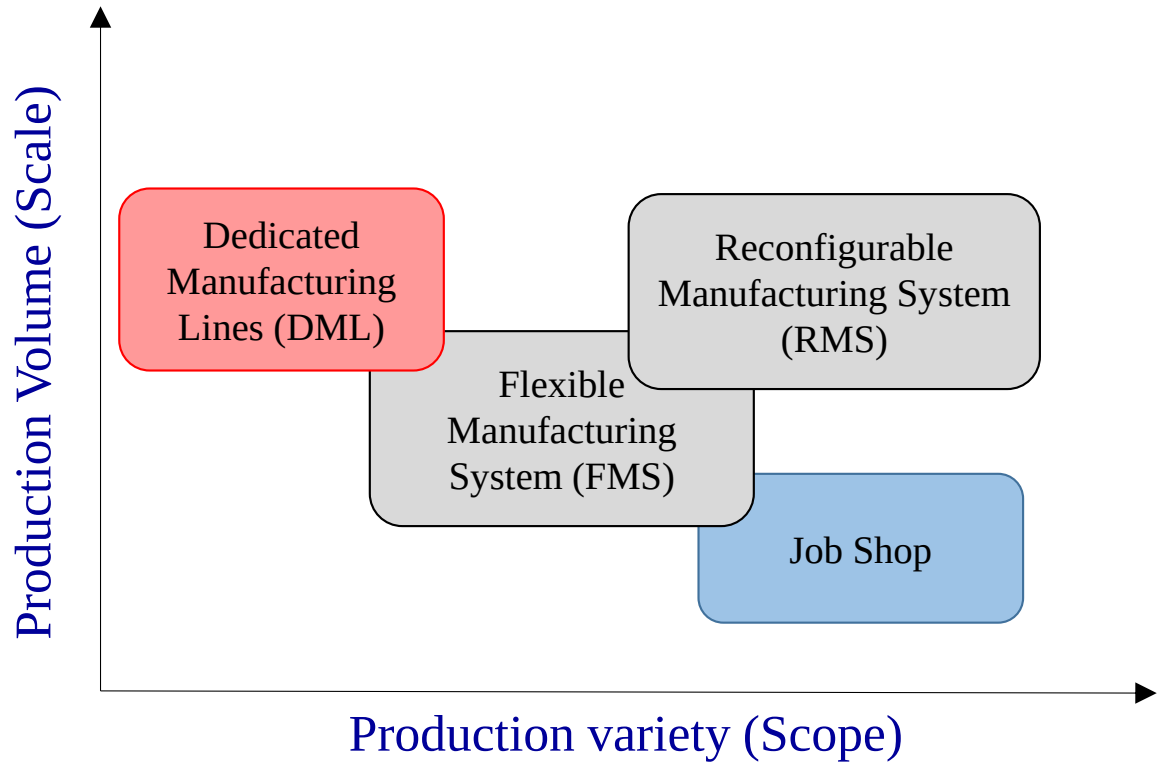

Figure 12 Manufacturing systems paradigms positioning.

The definitions and the objectives of the main three manufacturing systems paradigms, DML, FMS, and RMS, are summarized by (ElMaraghy 2005) in Table 3 below.

Table 3. Manufacturing systems paradigms (H. A. ElMaraghy, 2005).

Paradigm	Definitions and objectives
Dedicated manufacturing lines (DMLs)	• A machining system designed for the production of a specific part type at high volume. • Cost-effectiveness is the driver achieved through pre-planning and optimization.
Flexible manufacturing systems (FMSs)	• A Flexible Manufacturing System is an integrated system of machine modules and material handling equipment under computer control for the automatic random processing of palletized parts. • The objective is to cost-effectively manufacture several types of parts, within pre-defined part families that can change over time, with minimum changeover cost, on the same system at the required volume and quality.
Reconfigurable manufacturing systems (RMSs)	• A Reconfigurable Manufacturing System is designed for a fast change in structure to quickly adjust production functionality and capacity, within a part family, in response to changes in market requirements. • The objective is to provide precisely the functionality and capacity that is needed when it is required.

2.6 Cyber-Physical Systems (CPSs)

One of the main challenges of Industry 4.0 is CPSs (Chen 2017), which is defined by the US National Science Foundation (NSF) as the tight conjoining of and coordination between computational and physical resources (NSF 2018). Lee et al. investigated the challenges that faced CPSs and found that the main problem is the fact that the data generated from different sources have orientations. Current-day abstraction systems do not work when it comes to the calls and thread methods. Other challenges from the system-design and architecture perspective, specifically computer networking, make current manufacturing systems incompatible with CPS (Lee 2008). (Lv et al. 2019) applied a CPS model to an automated electric vehicle using a ML approach. The method optimized different CPS parameters to improve different performance objectives, such as dynamic vehicle performance, and drivability, along with varying styles of driving. The results validate the feasibility and effectiveness of the proposed CPS-based method for three driving techniques, which are aggressive, moderate, and conservative (Lv et al. 2019).

The new achievements in the information technology domain, especially in communication, wireless network, and cloud computing, help smart factories to benefit from the information and data that comes from different production sectors. Information from various manufacturing components, such as machines, and maintenance log files, can now all be integrated into one single system called CPS, which is the backbone of smart manufacturing. Advanced sensors and communication technologies allow for the monitoring of manufacturing components and the support of decision-making by using advanced data processing, mining, and analysis methods. Therefore, we can define CPS as the technologies that are used to manage the interconnected systems between the physical components and computational

resources (Baheti et al. 2011). While (Lee et al. 2013) introduced recent advances in manufacturing informatics related to big data, cyber-physical-systems, predictive manufacturing systems, and Industry 4.0 and stated that due to the lack of smart analytical tools, many manufacturing systems are not prepared to deal with big data because it involves more than just connecting machines to sensors or connecting machines to each other. It provides users with the vision needed to make improved decisions. They also enhanced the manufacturing information system with 5C roles, which are (1) connection (networks and sensor), (2) cloud (data availability), (3) content (meaning and correlation), (4) community (social and sharing), and (5) customization (value and personalization). (Lee et al. 2014) Introduced a cyber-physical system framework for self-aware and self-maintaining machines, which is defined as a system that can self-assess its degradation and strengths and make smart maintenance decisions to avoid possible problems. Because the CPS is still in the initial phases of expansion, it is important that we define CPS enablers. Cloud computing, IoT, and big data (see Figure 13) are examples of CPSs. Each of these terms is introduced in the following sections.

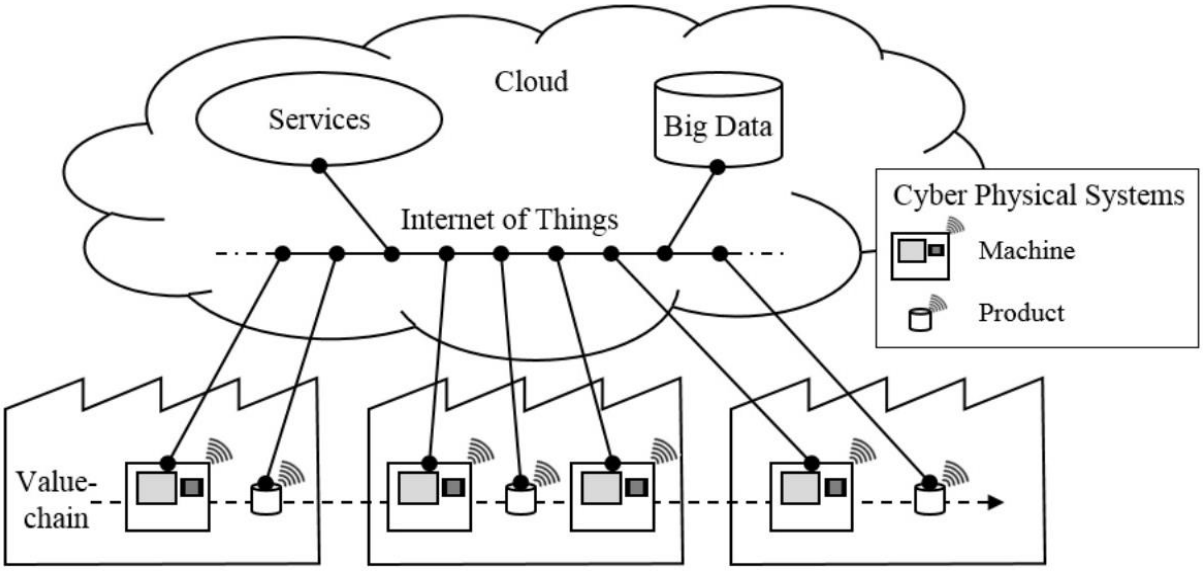

Figure 13. Cyber-Physical System structure (Gölzer et al., 2015).

2.7 Internet of Things (IoT)

Within the last decade, with the new technology enhancement in the domains of communication and sensors, there has been vast amounts of work in the fields of CPS and the IoT (Sun 2012, Li et al. 2018). A simple definition of the IoT is that it is a system in which objects with sensors in the physical space are connected to the internet through wired or wireless network connections (Babiceanu et al. 2016). It is highly expected that everything around us, like devices, mobile systems, and objects, will be connected through the IoT, which will transform our big world into one big network. These emerging technologies will enable not only human but also the automated decision makers to deal and communicate with physical equipment and grant them the capabilities of collecting, storing, sending, and receiving generated data (Mashal et al. 2015).

For the manufacturing domain, physical devices like sensors, cameras, readers, and actuators will be connected through the IoT. Together, the IoT and CPS are forming a state-of-the-art smart manufacturing system (Peruzzini et al. 2017). With the implementation of this system in manufacturing processes, a series of achievements and profits are expected to be made, such as process automation, efficiency increases, improved automatic control, improved maintenance, the exchange of information in real time for decision support, the ability to monitor overall manufacturing operations within the supply chain, and the ability to communicate and exchanging information within manufacturing facilities and the supply chain (Wang et al. 2015). Within the framework of the IoT in manufacturing, cyber-physical networks of smart devices are sensing, gathering, sending, and receiving data and considered to be the link between physical objects and the internet. Cyber-physical devices are also called "smart objects." They use several technologies that are merged, such as communication technologies, sensing technologies, internet protocols (IPs), and embedded devices, to form complete systems in which physical and digital worlds meet and continuously interact and form a new paradigm in the manufacturing field (Borgia 2014). As considered by (Qiu et al. 2015), the IoT provides a way to share information about all manufacturing operations at all available manufacturing levels in real time. The proposed model contains three parts: (1) physical devices, (2) information infrastructure, and (3) a DSS (decision support system).

2.8 Cloud Computing (CC)

The idea of cloud computing was introduced in the late 2000s, and it has been massively improved over the last few years as a result of the significant increase in the number of applications that require the cloud computing technology to operate. Sharing components and services is the key to cloud computing, which significantly contributes to Industry 4.0 (Moghaddam et al. 2018). The cloud computing concept assumes that everything is a service even it is processor resource as shown in Figure 14. That is why we talk about software are software as a service, or SaaS, in cloud computing. In cloud computing, we consider hardware as a service (HaaS), infrastructure as a service (IaaS), platform as a service (PaaS), and so on (Mikusz 2014).

Compared to other domains, the manufacturing domain was late in getting the benefit of the huge capabilities offered by cloud computing technology. In the last few years, researchers and manufacturers in the industry domain started to use this technology as one of the most significant enablers for the new paradigm of the fourth manufacturing revolution of Industry 4.0 (Mourtzis et al. 2014).

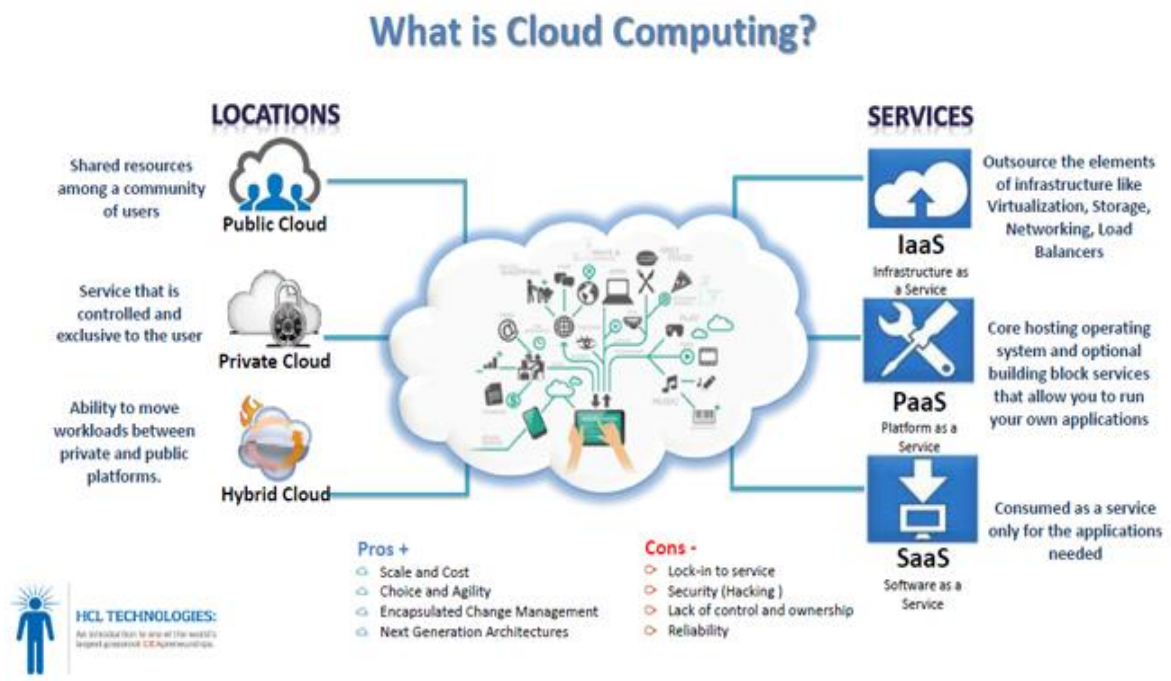

Figure 14. Cloud Computing ("Cloud Computing," 2016).

Therefore, it is essential for researchers and manufacturers to have a clear definition of cloud manufacturing. Hence, during the literature survey, two definitions were found. Firstly, (Li et al. 2010) defined cloud manufacturing as a "computing and service-oriented manufacturing model developed from existing advanced manufacturing models and enterprise information technologies under the support of cloud computing, the Internet of Things, virtualization and service-oriented technologies, and advanced computing technologies." On the contrary, the second definition gives more information about the processes and resources than the first. It was written by (Xu 2012), who defined cloud manufacturing as a "model for enabling convenient, ubiquitous, on-demand network contact to a shared pool of configurable manufacturing resources (e.g., manufacturing software tools, manufacturing equipment, and manufacturing capabilities) that can be rapidly provisioned and released with minimal management effort or service provider interaction." This enables users to access the cloud whenever required and request services at any stage of the product lifecycle, starting from design, development and manufacturing to testing and even management (Xu 2012). With an increased concentration on using cloud technology in both design and manufacturing activities, a new combined process called cloud-based design and manufacturing (CBDM) was proposed by (Wu et al. 2015). This new CBDM concept considers the access to the shared distributed manufacturing facilities and resources within the reconfigurable manufacturing lines, which is expected to have a significant productivity improvement, such as improvements to production efficiency, the ideal allocation for resources, and last but not least, reductions on overall product lifecycle costs. The CBDM system and any other similar cloud manufacturing concept in the same domain are expected to have a specific computing concept and component, such as the following (Wu et al. 2015):

- CPS collects real-time data to be stored on the cloud.
- Controlling and monitoring the whole production process remotely.
- Resources to be shared and utilized.
- Everything as a service XaaS, for example, SaaS, PaaS, HaaS, and IaaS services.
- Having global access for the gathered data by the cloud-based distributed file system.
- Having high capabilities for communication and Sharing information like what we have in social media.
- Having high capabilities to store, analyze, and process big data with efficient search techniques.

2.9 Big Data Analytics for Healthcare

The rapid growth and expansion of digital healthcare data started to play a key role in the development of healthcare research and operations. This secured a variety of tools that collect, analyze, manage, and integrate large amounts of different structured and unstructured data generated by existing healthcare systems. The healthcare aspects such as care delivery, disease exploration, and personalized medication, can be improved by utilizing the BDA, but the research development and enhancement rate in this domain is still delayed by some vital issues and challenges that limit the success due to the limited capabilities of handling the vast amount of complex data that has a high velocity, high volume, and high variety. (Fang et al. 2016) Proposed the key techniques, challenges, and future guidelines for the BDA in health informatics and summarized the main challenges into four Vs (velocity, volume, veracity, and variety). They also proposed efficient data-processing techniques for BDA in healthcare, such as data collection, storage, analysis, and decision-making support.

Meanwhile (Belle et al. 2015) proposed the key challenges involved in the implementation of BDA in healthcare, with a focus on some specific data types, such as image, signal, and genomics data. (Dimitrov 2016) reviewed the medical internet of things (mIoT) and BDA in the healthcare domain, which introduces the digital health adviser (personalized health coaches) utilizing the new technologies such as devices that monitor health indicators frequently, devices that auto manages treatments or devices that monitor real-time health information when a patient self-administers a therapy.

CHAPTER 3. RESEARCH METHODOLOGY

3.1 Overview

To achieve the objective mentioned above, big data analytics tools, varying data science approaches, such as machine learning, artificial intelligence, deep learning, and neural networks, statistical association rule discovery models, and experiment design tools, such as factorial design, are applied. Advanced data structures, such as the graphical databases used in social media applications, will be employed to allow for the efficient handling and analysis of vast amounts of complex systems data. The Spark Ecosystem and multi-agent systems (MASs) can be employed. The state-of-the-art reconfigurable manufacturing system (iFactory) available at the IMS centre will be utilized in testing and verifying the developed methodology.

Usually, for data that is not a big data, data analysts are able to complete data mining and machine learning using popular standard, tools such as Weka, Python, and R (Pedregosa et al. 2011) (Hall et al. 2009), which are popular languages for data analytics due to the big number of components that are ready and available to help in solving their data issues. The ability of these standard tools is limited by data size, type, and processing capabilities because they can only be used to process structured data on single machines. Other machines are required to handle large volumes of data to form multi-core processors or more powerful machines. However, moving data from one machine to another is time-consuming. Therefore, Apache Spark (Zaharia et al. 2016) is one of the most actively developed open source ecosystems among data tools, big or small. It is used for high volumes and varieties of data (big data). To help to address these issues, Spark provides data analysts with a great, integrated engine that is both fast and easy to use, which is actively increasing the handling performance in the machine learning processing (streaming, and real-time query handling) at a high scale. Data specialists are spending most of their time preparing their data instead of building the needed models to solve their data issues. To help solve this issue, Spark provides libraries for machine learning that simplify, scale, and easily integrate with the other tools, such as those in the Hadoop Ecosystem and MAS (Joseph Bradley 2016).

In the following sections, I explain the research plan and the methodologies used in both manufacturing systems and healthcare systems.

3.2 Research Plan

The research plan shown in Figure 15 shows the commonalities and differences between the two domains (manufacturing and healthcare). Both areas start with the data mining stage to extract problem

classes, and this is followed by labelling all data with the discovered classes. Then the needed data preprocessing is implemented to prepare the datasets for the next phase, which is features handling. Feature handling is the main difference between the two domains because, in manufacturing systems, feature extraction is implemented as convolutional neural networks (CNN) and utilized for extracting the unknown features map while in the healthcare domain, the features are already known and the main objective is to select only the highly-relevant features. Therefore, ranker and wrapper feature selection methodologies are implemented. Finally, the prediction model and data classification are developed for both domains utilizing the proper techniques. In manufacturing systems, the developed model for defect detection using CNN is presented as a case study number one, which is detailed in chapter four, section two in this work. Meanwhile, for healthcare, two models for personalized breast cancer treatment are developed. One is presented as a case study number two (using supervised machine learning), which is detailed in chapter five, section two, and the other one is presented as a case study number three (using semi-supervised machine learning), which is detailed in chapter five, section three.

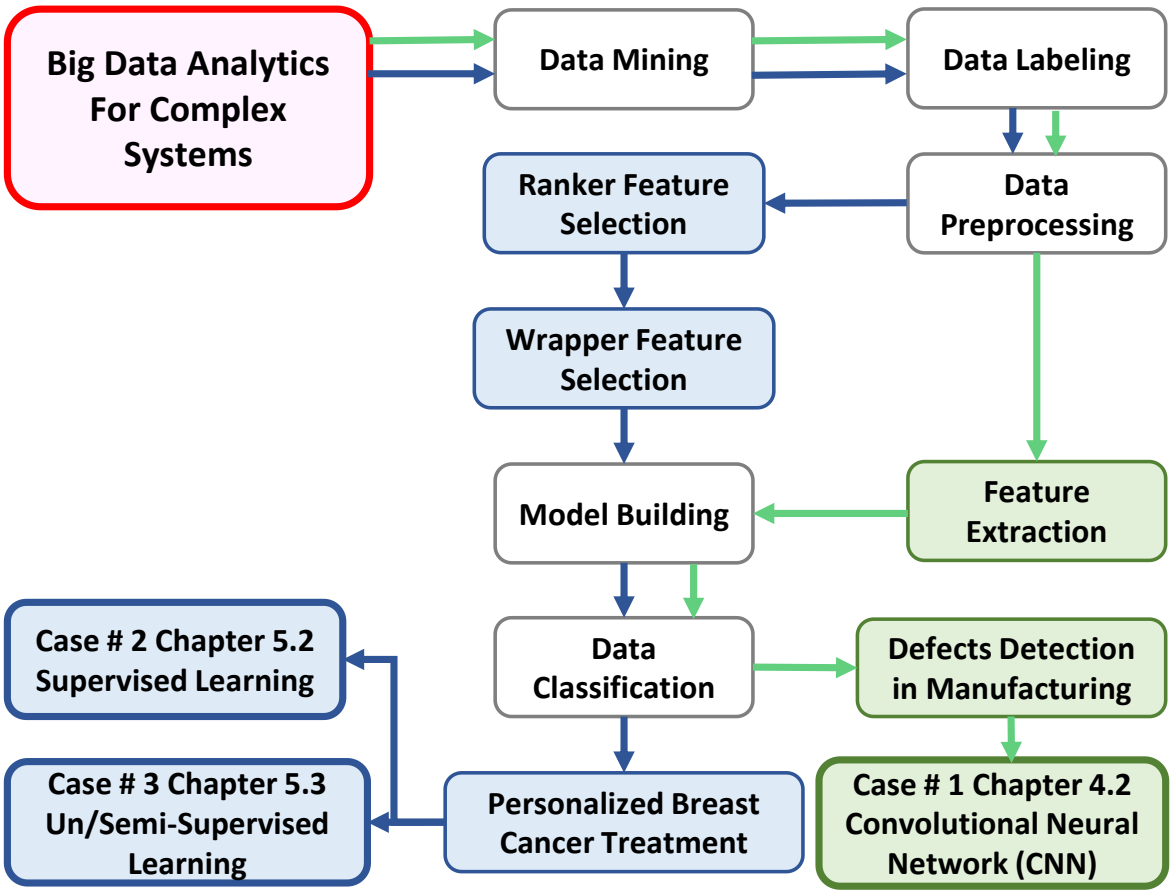

Figure 15. Research plan.

The following part shown in Figure 16 is the high-level IDEF0 graphical function model for the big data analytics of complex systems (in both domains). It illustrates the elements of the proposed research, which include inputs, outputs, mechanisms, and controls. The inputs are the big data sources, and the

outputs are the prediction outputs and the decision support actions. Utilizing all available mechanisms, such as big data tools, supercomputers, and IoT.

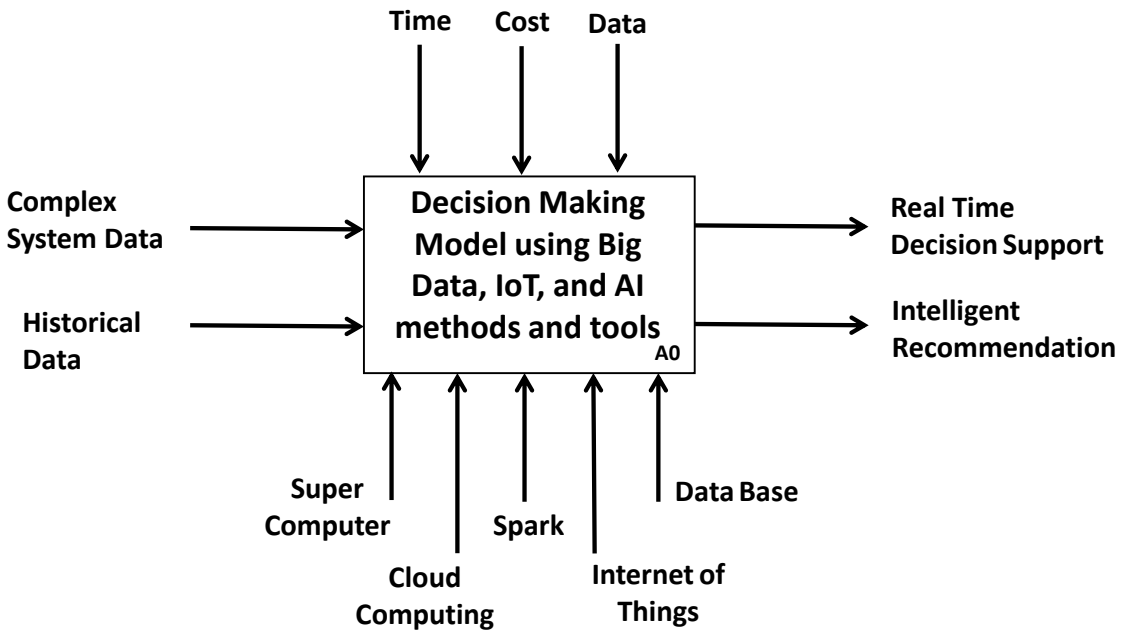

Figure 16. IDEF0 for big data analytics for both domains.

The following IDEF0 shown in Figure 17 is the Detailed IDEF0 for big data analytics for complex systems (in both domains). It starts with data mining to study the system data and the historical data based on the features to identify the problem classes. Then labelling all the available input data with the figured classes is done. To create a classification model, the data should be normalized. Therefore, the data preprocessing stage is required to have normalized classes ready for the prediction model by scaling, resizing, and resampling the input data. This is followed by ether feature extraction or feature selection to limit the number of features to only the high-relevant features that are used to build the prediction model. The final stage is the data classification stage that is responsible for the final real-time decision making after having the needed and proper training, validating, testing, and optimizing the prediction model.

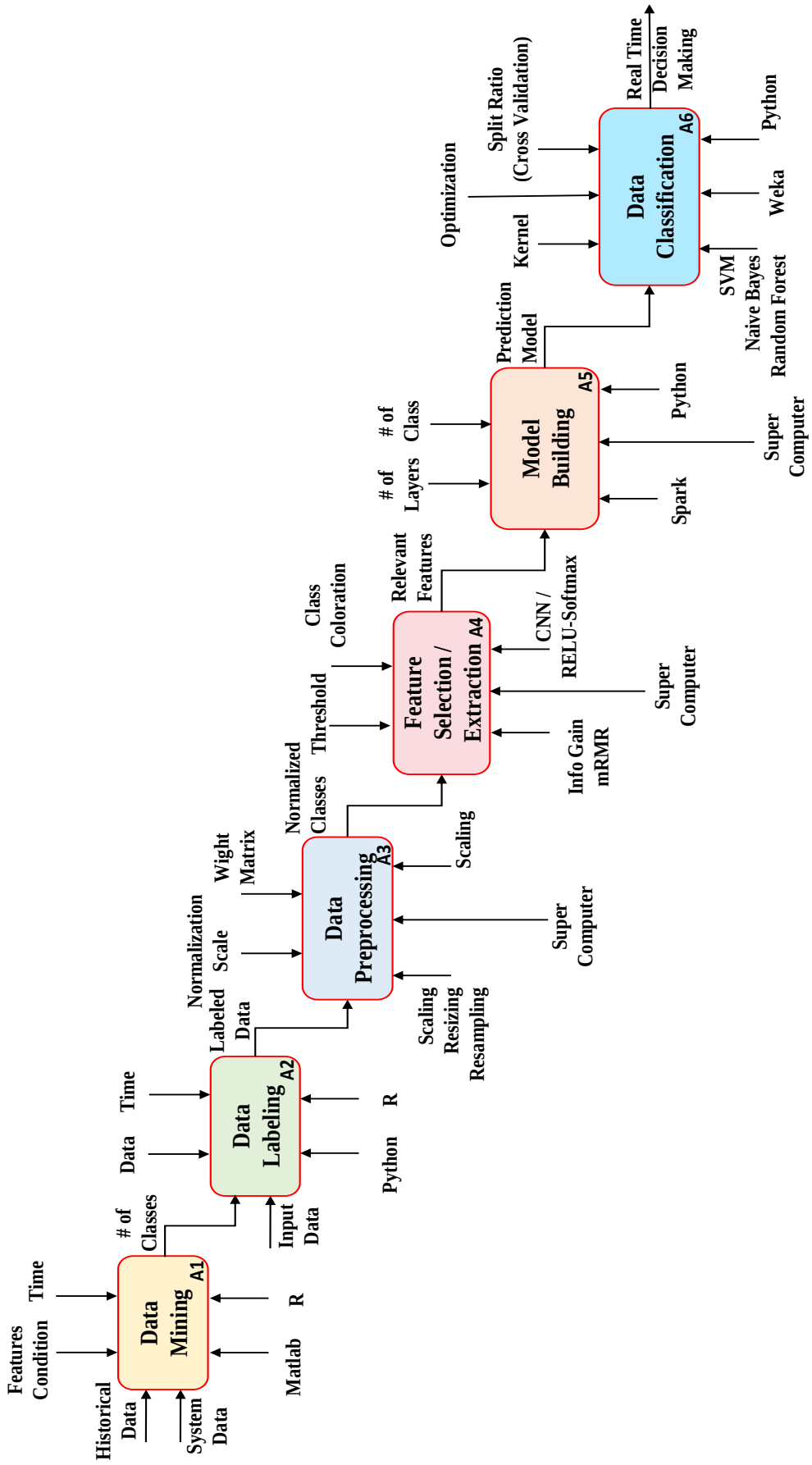

Figure 17. Detailed IDEF0 for big data analytics for both domains.

Another way to represent this research is through the Zachman framework (Zachman 1987), which is a holistic design approach. This method capsulate the architecture and the foundation in large complex design activities. The framework is concerned with answering many questions, like what, when, who, how, where, and why for each step of the activities, starting with the enterprise model (conceptual) and moving to the system model (logical), and finally the technology model (physical). Figure 18 below shows the detailed framework.

Model	Data What	Function How	Network Where	People Who	Time When	Drivers Why
Enterprise Model Conceptual	Semantic Data Model	Research Resources and Facilities	University of Windsor	Research Team	Ph.D. Period	Complex Systems
System Model Logical	Logical Data Model Embodiment Design	IMSC Lab, Hardware, and Software	Engineering CEI	Student (Me) Committee	3 to 5 Years	Manufacturing and Health Systems
Technology Model Physical	Detailed Design Data Model	Spark, iFactory, and Statistics	IMSC Lab	Student (Me) Supervisor	Summer 15 summer 19	Big Data Analytics tools

Figure 18. Zachman framework.

3.3 Manufacturing System

In this section, the workflow of the manufacturing system model is depicted as follows:

3.3.1 Big Data Predictive Analytics Workflow

The workflow for the big data analytics predictive model is shown in Figure 19 starts with the data sources as an input for the model, such as historical data, data streams, and the data extracted from the system. This data should be validated for any missing or invalid values or even outliers. After the validation process, the data should be preprocessed using many techniques, such as normalization, augmentation, and filtering. This ensures that we have clean, valuable data ready for use. Then we apply the predictive model that will assist in decision making utilizing AI and machine learning methodologies, such as CNN and supervised, unsupervised, and semi-supervised machine learning. The predictive model sometimes requires additional data processing, such as data scaling, data resizing, and data resampling, to deal with issues such as over sampling and under sampling. Finally, for improved representation, the output decisions will be stored, and the results can be represented using data visualizations methods.

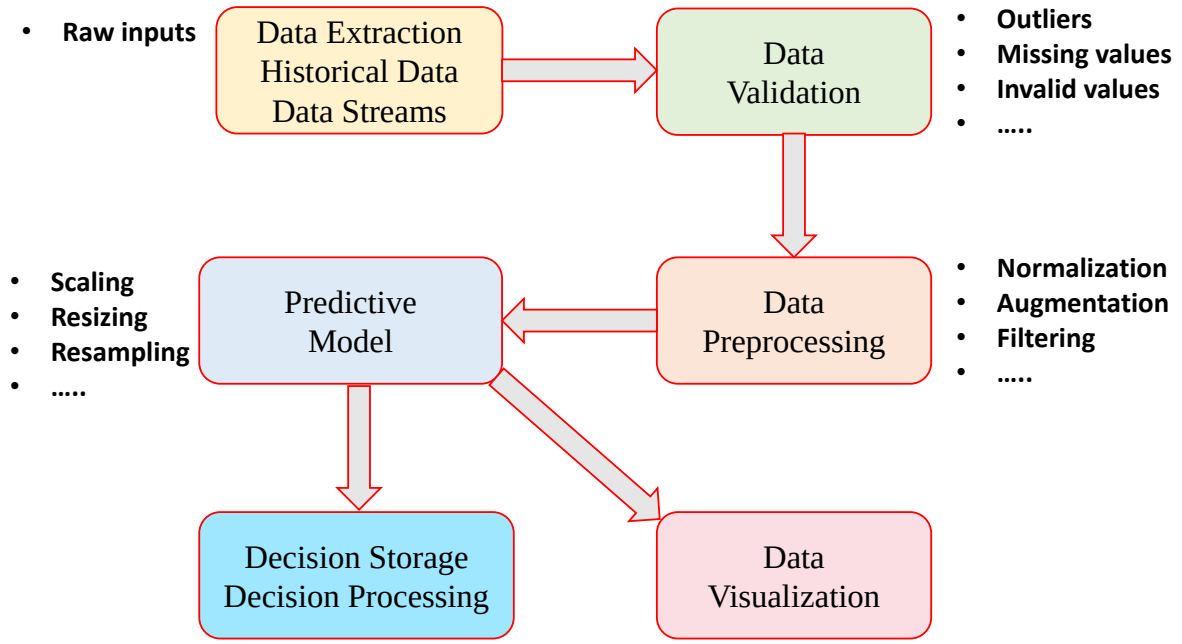

Figure 19. Big Data analytics predictive workflow.

3.3.2 IDEF0 for Manufacturing System

The following part is the high-level IDEF0 graphical function modelling for smart manufacturing systems shown in *Figure 20*. It describes the main activities of the proposed research in the manufacturing system domain, including all the inputs, outputs, mechanisms, and controls. The inputs are the big data sources, such as sensor data, system data, historical data, and log files, and the outputs are the error fixes, quality checks, and decision support actions. Utilizing all available mechanisms, such as big data tools, supercomputer, CPS, and IoT.

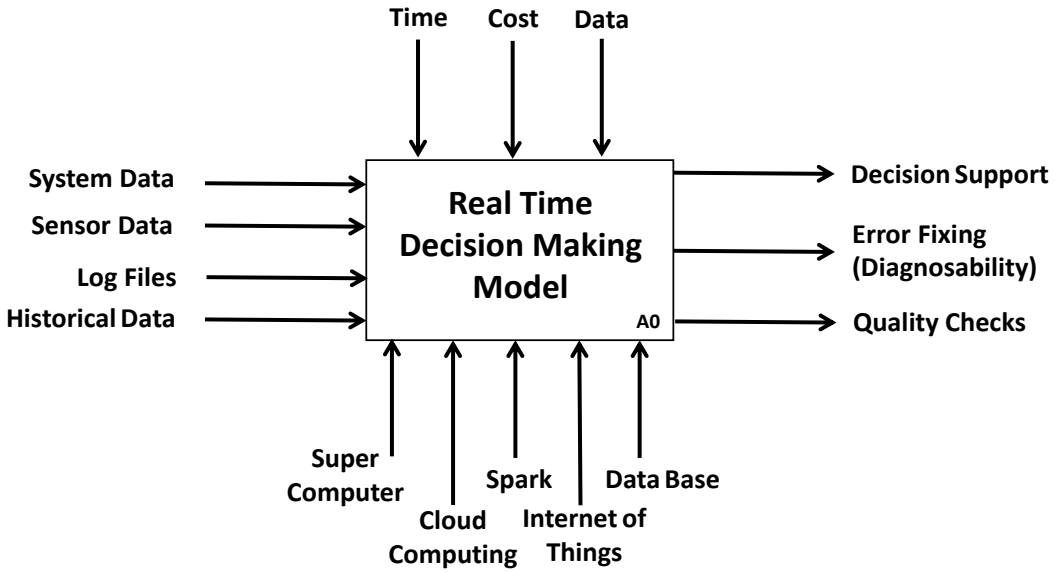

Figure 20 IDEF0 for smart manufacturing system.

The detailed IDEF0 for this part of the research is shown below in Figure 21. It starts with data mining to analyze the input data, such as sensor data, historical data, and log files. Considering the product status (good or bad) to figure out the problem classes, which are defected or non-defected products using the product images. Then the product images are labelled as 1 for non-defected products or 0 for defected products. A few thousand images are captured by the inspection station camera, but that is not enough for training the model. Therefore, the next step is to apply data preprocessing augmentation to increase the number of images from only 5,000 images to 20,000 images, 10,000 images of each class. This is done by flipping, rotating, and resizing the images. Then applying the CNN to utilize transfer learning by implementing the resnet-50 and a 3 by 3 kernel matrix with one stride, and using the RELU and SoftMax activation functions to extract a features map. Then the prediction neural network model is built using sickitLearn and TensorFlow libraries to build a fully-connected specific number of layers. Finally, to classify the data and make a decision, either the product is good or bad.

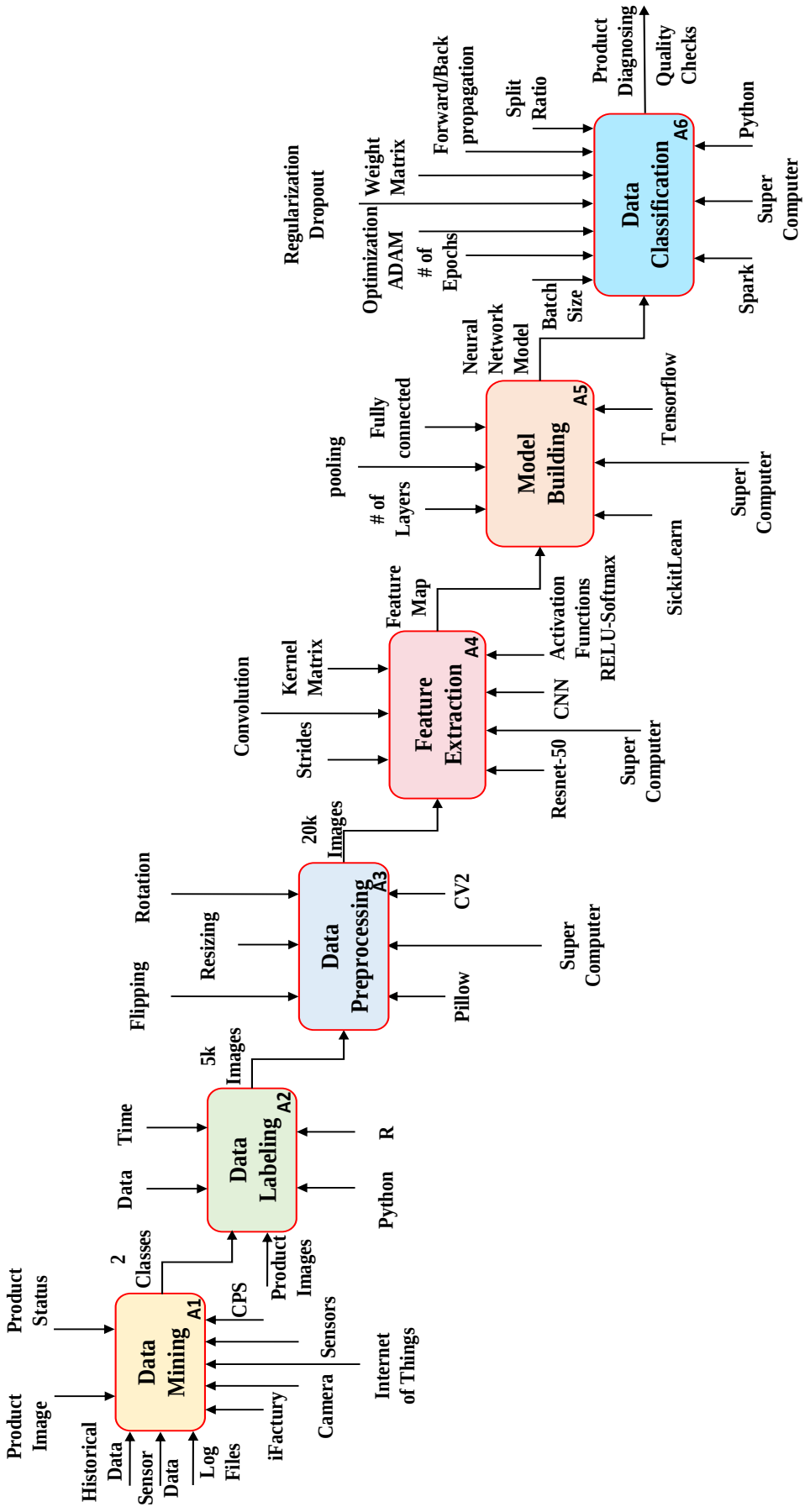

Figure 21. Detailed IDEF0 for the smart manufacturing case study.

There are several steps within the data classification part, as shown in Figure 22 below. Starting with the model training using a split ratio of 70% for model training, 20% for model validation, and 10% for model testing. The training phase involves the utilization of forward/backpropagation to improve the learning rate by changing the weight matrix. This step is followed by model tuning by utilizing the batch size and the number of epochs, model optimization using the ADAM optimizer, model regularization using deferent dropouts, and finally, validating and testing the model to ensure that it is neither over-fitted nor under-fitted before applying the final classification to the not-seen sample (samples not used in training nor in the validation) for decision making.

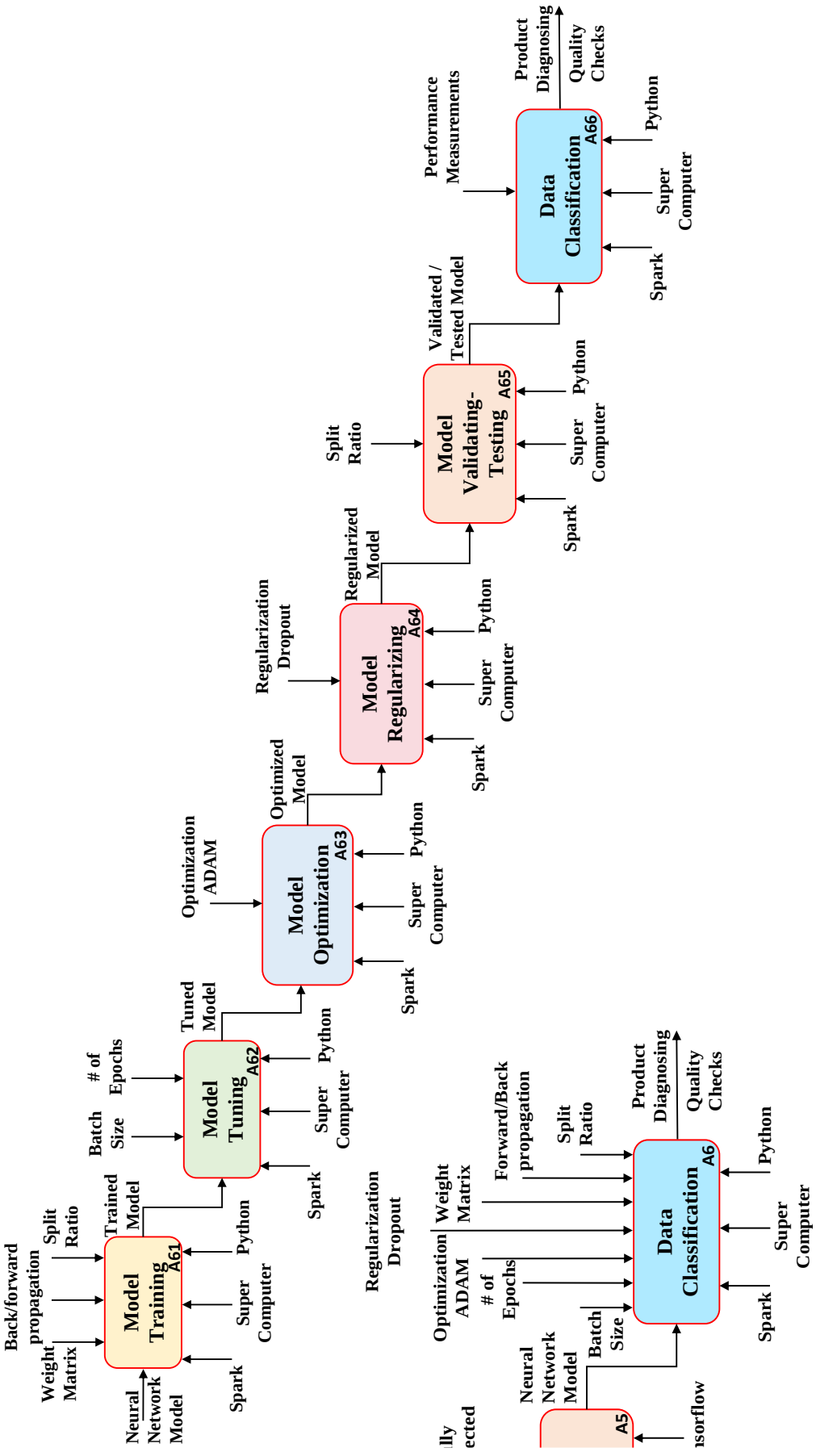

Figure 22. IDEF0 for data classification part.

3.3.3 Convolutional Neural Network (CNN)

CNN is a typical application of deep neural networks. They consist of input layers, multiple hidden layers, and output layers. Typically, the hidden layers of a CNN consist of multiple convolutional layers, activation functions, pooling, fully-connected, and normalization layers (see Figure 23). The CNN proves itself with reliable performance in processing media files, such as images and videos (Liu et al. 2017). CNN requires high computational power. It works very well with GPUs and can utilize supercomputer resources in such a way that the training phase is highly efficient. Nowadays, CNN is among the best learning tools that have been utilized in many fields, such as handwriting recognition, facial recognition, image recognition, and speech recognition, to name just a few.

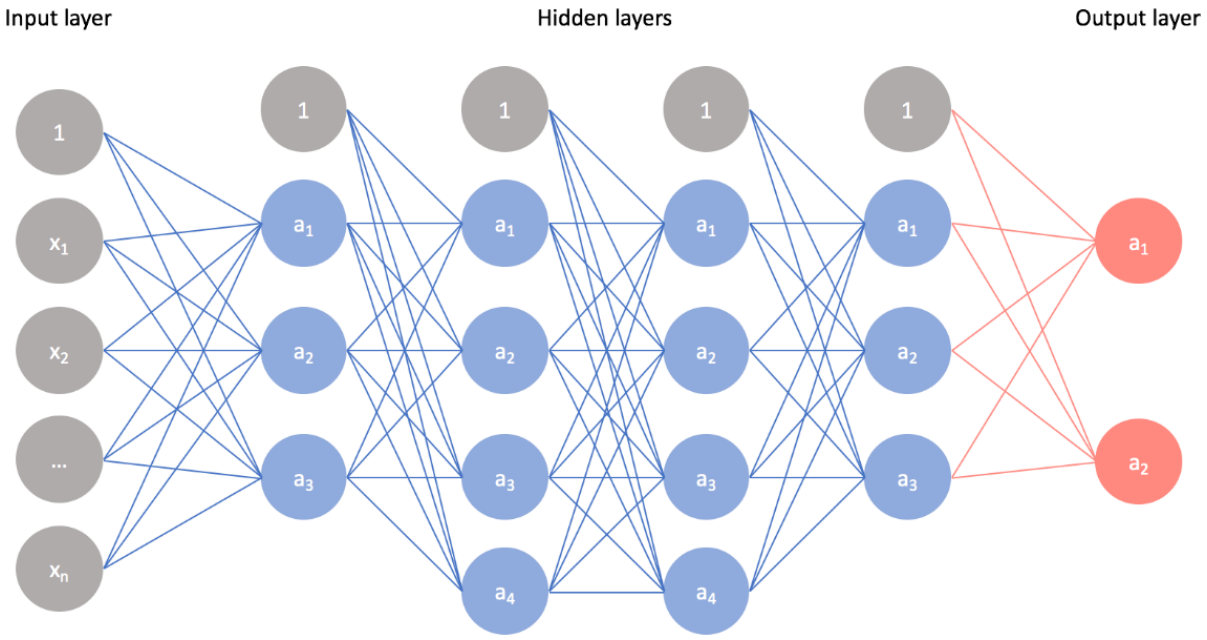

Figure 23. Convolution neural network (CNN) layers (Jermyjordan 2019).

A CNN is a multi-layer structure that is built with two types of layers, convolutional layers and subsampling layers that are connected alternatively (Goodfellow et al. 2016). Generally, the convolution layer's function is to extract features from the 2D data. Then these features are mapped in the sampling layers. Similar to a NN, CNN uses an activation function to achieve scale invariance.

The training procedure for CNN is similar to that of a standard NN. The first step is the feed-forward propagation of the information to extract features by applying filters. In the second step, the error between the expected and actual values is calculated, and backpropagation is done to minimize the error (Bengio 2009).

3.4 Healthcare

In this section, the workflow of the health informatics model is depicted as follows:

3.4.1 IDEF0 for Healthcare

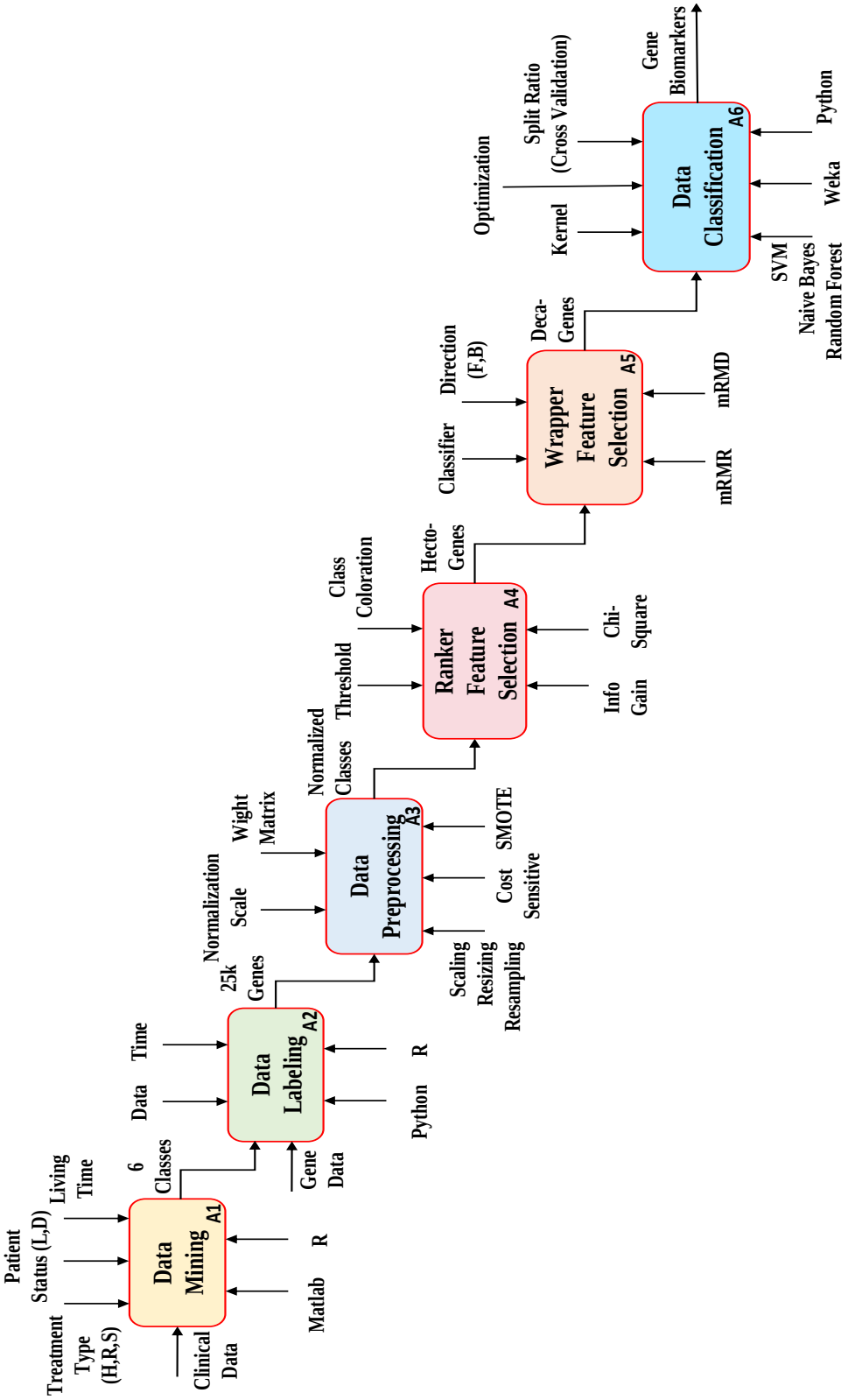

Figure 24. IDEF0 for the healthcare section (gene biomarkers).

Above is the IDEF0 for the healthcare section of this research Figure 24. It starts with the raw data from the patients' clinical data records. After an intensive investigation, the model was designed based on the clinical data to determine which health informatics problem will be studied. After completing the proper data mining, the model idea, which is the treatment survivability for five-year interval survival patients, came up, and six classes of records were identified. The second phase is to label the patient's gene expression samples with the six new classes from phase one. Each sample has about 25,000 gene expressions. This will result in the accumulation of about 2,433 data points about patient clinical information and about 25,000 gene expressions for each patient, which will give more than 60,000,000 gene expressions.

This model uses a one-versus-rest algorithm to handle the multi-class problem. This leads to an imbalanced class dataset. Therefore, we applied several data preprocessing techniques to handle this issue, including oversampling with synthetic data, resampling, cost-sensitive classification, and SMOTE. This ensured that the data was normalized and ready for the next stage.

This massive number of features will be handled by two stages of feature selection. The first one is using ranker (filter) feature selection, such as Info-Gain, as given in Equation 2, which checks whether the correlation of the class with the most relevant features is higher than a specific threshold and removes redundancy, noise, and irrelevant features. This will reduce the number of features from about 25,000 genes to around a few hecto-genes, which is still too many, and it ignores the interactions between the features. That is why the second method of feature selection is applied. The second method is the Wrapper feature selection, whereby the relations between features are considered using a greedy learning algorithm, which removes or adds one feature at each stage to best improve (or least degrade) the cost function. It is using either backward or forward elimination. Backward elimination starts with all features (n, n-1, n-2, …, 1) and progressively remove features one by one (never adding any) while forward elimination starts with an empty set (1, …, n-2, n-1, n) and progressively adds features one by one (never removing any). After applying the wrapper feature selection, the number will be reduced from hecto-genes (hundreds) to a deca-genes (tens), or even less.

$$IG\big((X|Y)\big) = H(X) - H((X|Y)) \quad \text{Where X is the set of features that maximize the entropy of the class vector Y.} \tag{2}$$

Finally, the last step is the classification stage whereby the gene biomarkers are used to classify the different classes using well-known classifiers, such as support vector machine (SVM), naïve Bayes, and random forest — utilizing split ratio as 10-fold cross-validation. Kernel techniques are used to implicitly transfer the data to a higher dimension where the data is separable. These techniques are better known as kernel tricks. Optimizers are used to handle overfitting and lead to improved performance efficiency.

Dropout and ADAM techniques are used in this phase. A final result is a small number of biomarker genes for predicting proper treatment therapy for the patient based on his or her genes.

As noticed, the core part of the final step is the classification process. As mentioned above, the following several classifiers are utilized.

3.4.2 Naive Bayes classifier

The Naive Bayes method is a supervised learning classification method based on Bayes' theorem and the conditional independence assumption (Mitchell 2005). For a given training data set $T = ((x_1, y_1), (x_2, y_2), \dots \dots (x_i, y_i))$ with label $y, y_i = c_1 , c_2, \dots \dots \ c_k \ , \ i = 1,2, \dots \dots \ N$ assume there are S_l possible values for $x^l, l = 1,2, \dots \dots \ n$; and there are K possible values for Y. Naïve Bayes first learns the joint probability distribution $P(X, Y)$ of the input and output by the conditional probability distribution based on the conditional independence assumption:

$$P\big(X = x \big| Y = c_j\big) = P\big(X^{(1)} = x^{(1)}, \dots \dots, X^{(n)}\big) = \ x^{(n)} | \ Y = c_j$$
$$= \prod_{l=1}^{n} \mathrm{P}\big(\mathrm{X}^{(l)} = \mathrm{x}^{(l)}\big) \ \mathrm{j} = 1,2, \dots \mathrm{K} \qquad (3)$$

Then, based on the learned model, the output label y with the biggest posterior probability for the given input x can be calculated via Bayes' Theorem:

$$P\big(Y = c_j \big| X = x\big) = \frac{P\big(X = x \big| Y = c_j\big) \, P(Y = c_j)}{\sum_j P\big(X = x \big| Y = c_j\big) \, P(Y = c_j)} \qquad (4)$$

And

$$y = argmax_{c_j} P\big(Y = c_j\big) \prod_l P(X^{(l)} = x^{(l)} | Y = c_j) \qquad (5)$$

The use of a naïve Bayes classifier is a common method for classification because it is easy to implement with high efficiency.

3.4.3 Support Vector Machine (SVM)

SVM is a computational supervised learning method for small sample classification (Widodo et al. 2007). Algorithmically, SVM builds the best separating hyperplane $f(x) = 0$ between datasets by

solving a constrained quadratic optimization problem. To do so, based on the structural risk minimization (SRM) (Schölkopf et al. 2002).

$$y = f(x) = W^T + b = \sum_{i=1}^{N} Wi\ xi\ + b \qquad (6)$$

where W is an N-dimensional vector and b is a scalar. The optimal separating hyperplane is the separating hyperplane that creates the maximum distance between the plane and the nearest data, that is, the maximum margin, as shown in Figure 25. By converting the optimization problem with Kuhn-Tucker condition into the equivalent Lagrangian dual quadratic optimization problem, the classifier based on the support vector can be obtained.

SVM has been widely accepted as a reliable classifier. It can be used to incorporate different kinds of kernels to implicitly move data into a higher dimension where the samples points are more separated. The idea behind implicitly transferring data into a higher dimension using dot products is to avoid the use of costly transferring functions (Liu et al. 2018).

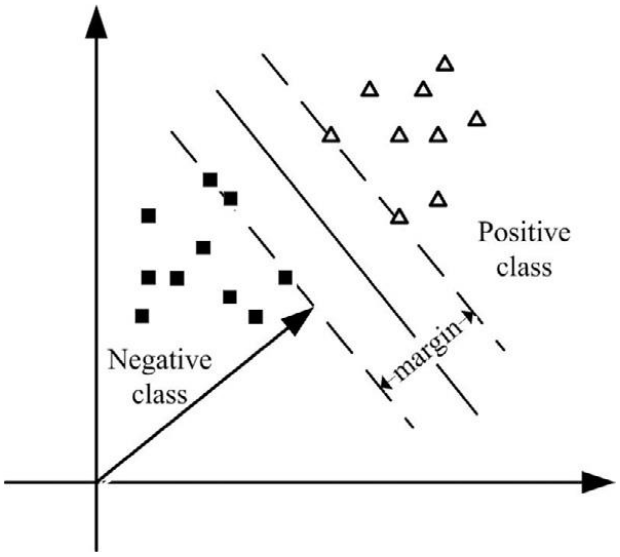

Figure 25. The optimal hyperplane for binary classification by SVM (Liu et al. 2018).

CHAPTER 4. MANUFACTURING SYSTEMS APPLICATION

4.1 Overview

In a modern manufacturing facility, a vast amount of different types and formats of data, like structured, unstructured, and semi-structured, are being generated continuously. Big-data analytics is a new area of computational intelligence that offers new theories, techniques, and tools for processing and analyzing large datasets. It is a discipline of growing interest and importance. Data mining can provide significant competitive advantages to manufacturing organizations by exploiting the potential of large data warehouses. Thus, our objective in this part of the research is to develop a new method or tool for gathering, analyzing, and processing big data (structured or non-structured) that is generated through Industry 4.0 systems (e.g., machine sensors or log files) that support manufacturing system lifecycle management from conception to actual manufacturing through advanced knowledge discovery tools. The new tool will also support manufacturers' decisions sustainably and economically in designing and operating their manufacturing systems. Existing methods were not developed to handle big data. The proposed method will be able to handle big data by using modern data structures and data storage methods, such as graphical databases, and develop data-driven methodologies to solve manufacturing problems using mining techniques (Roy et al. 2014).

Some benefits of using big-data analytics techniques are shown in Figure 26 below, such as improving preventive maintenance, enabling condition monitoring to manage equipment effectiveness, and increasing manufacturing efficiency while reducing material and energy consumption (NIST 2018).

- **5%** reduction in energy costs.
- **10%** reduction in water consumption.
- **5%** decrease in batch cycle time.
- **10%** improvement in machine reliability.

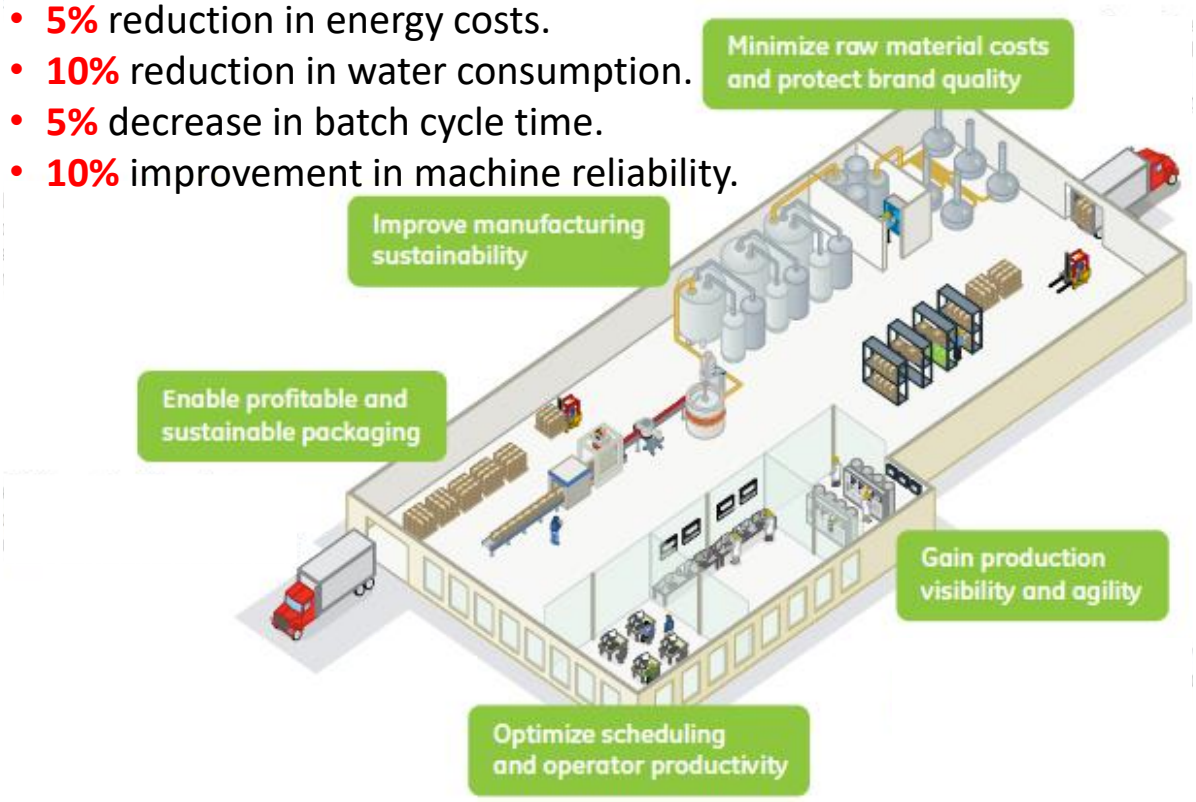

Figure 26. Benefits of big-data analytics techniques (NIST, 2018).

4.2 Case Study 1

Big Data Analytics for Defects Detection in Manufacturing Systems (Industry 4.0). (Abou Tabl et al. 2019)

4.2.1 Abstract

Modern manufacturing systems generate a large amount of data. Current trends in smart data analytics include the extraction of information from big data to make real-time manufacturing and operational decisions. This researcher presents a new model for smart decision-making for complex manufacturing systems. The model was developed and implemented on the iFactory, a reconfigurable smart manufacturing system located in the Intelligent Manufacturing Systems Centre (IMSC) at the University of Windsor. This learning factory, which is Industry 4.0 ready, includes many of the needed cyber-physical components and sensors. To implement smart (Industry 4.0) manufacturing IoT enablers, the sensor readings are streamed through a cloud supercomputer, and a big-data analytics learning model is applied for real-time decision-making. Big datasets of product image were gathered from the product inspection station. The IoT is represented by the reading of the individual product serial numbers using a camera as the sensor. Cloud computing is utilized in two ways: the first one consists of holding a large amount of data that is used to train the model, and the second is the execution of the model. A Compute Canada supercomputer was utilized for storing big data and executing the model to test it. To achieve satisfactory performance regarding the execution speed, it was necessary to implement a parallel architecture over a set of nodes with several cores for each. The number of nodes and cores can be configured based on the amount of data and the required decision-making speed. In this case, four nodes with 16 cores each, with 100 GB of memory per CPU (core) were utilized to hold the data and run a specially-developed convolution neural network (CNN) model. The CNN model was implemented to extract the features from the images and learn at the same time. The model was tested in a real-time environment and achieved a decision accuracy of up to 96.75%.

4.2.2 Problem Definition

Reducing the deep learning training time and achieving timely prediction with big data are significant problems that deter scientists from considering the use of artificial intelligence (AI) in their complex systems. This case study sheds light on the use of deep learning with transfer learning to accelerate the training time and achieve accurate prediction. This chapter is focused on the use of deep learning in a complex system to detect products with defects and sort them using a robotic system based on decisions made by an AI application. The dataset that was gathered for this work consists of 20,000 images for line products. The products are different variants of a specific product. The dataset is divided into 10,000 normal images and 10, 000 defected images. The images are labelled as normal or defected, as denoted by 0 and 1, respectively. The objective is to develop a real-time decision-making prediction model that

is scalable, reliable, and fault-tolerant to make the iFactory model Industry 4.0 compatible. This was done by applying the main Industry 4.0 key enablers, such as big-data analytics, CPSs, cloud computing, and the IoT.

4.2.3 Introduction

In the current digital transformation industrial era, Industry 4.0 has been receiving researchers' attention with the goal of automating production line decisions. The main idea is to utilize revolutionary AI capabilities, big-data availability, and deep learning technologies on production lines to diagnose any incident and take proper action. Current manufacturing complex systems (MCS) have been widely used to reduce the setup costs of restructuring production lines. However, the current MCS modules are not based on AI and, hence, cannot learn and adapt to the modern industrial environment to make a real-time decision. Moreover, MCS does not employ cloud computing, big data, or the IoT, which limits its ability to be expanded and used beyond its restricted design. To achieve this, the main modules must connect to an online server (supercomputer) that has big-data analytics and machine-learning capabilities.

In this research, we considered iFactory, which is a state of art learning factory that is located in an intelligent manufacturing system laboratory, which is one-of-a-kind of North America (ElMaraghy et al. 2012), as shown in Figure 27. It demonstrates the innovative physical integration of the different product lifecycle phases: iDesign, iFactory, and iPlan. The iFactory module in the IMS lab provides engineers, students, and researchers with a practical learning and training experience and offers a high possibility to transfer a concrete outcome to the industry. iFactory is an application of a reconfigurable manufacturing system (RMS). In this study, we enhance the iFactory module to fully implement Industry 4.0 enablers by streaming the sensor readings through the IoT and the cloud to a big-data analytics learning model for real-time decision-making. Based on lab-made learning modules that are illustrated in the case studies, we improved iFactory skills to monitor, diagnose, and predict unusual events, such as product defects. iFactory is used as a base model and integrated with the capabilities of AI, big data, the IoT, and could computing.

Figure 27. iFactory at IMS lab, University of Windsor.

Because big data is characterized by the three Vs (volume, velocity, and variety), the use case of iFactory will cover these three factors. The data generated from the sensors, cameras, and log files exist in differing formats, which achieves the product variety point. At the same time, the fast data generation from the sensors and log files demonstrates that the data is generated at a high velocity. Moreover, the vast amount of data generated by the sensors and cameras shows the volume characteristic.

In this work, a new intelligent iFactory that satisfies the need of Industry 4.0 elements and the ability to train from provided data, run on the cloud, be equipped with IoT sensing devices, and return real-time decisions are designed. The model consists of three components: An IoT component, where the data is acquired using a camera for capturing product images and their serial numbers to identify each product. The deep learning component is the brain of the system, whereby the system is trained and makes decisions. The procedural component is the one in which the decision-making will be performed to segregate normal or defected products.

This chapter has the following parts: a background section that includes a survey of the related literature, a materials and methods section with an explanation of the data gathering and preprocessing steps, a model architecture section about the methods and techniques, a results section, which includes performance and accuracy measurement results, and finally, a conclusion section.

4.2.4 Background

In today's competitive manufacturing environment, manufacturers are facing challenges in dealing with big-data issues, such as rapid decision-making for improved productivity, quality, and system sustainability. Manufacturing systems are not ready to manage and obtain the benefits of big data due to the lack of decision-making methods and tools. Manufacturers are forced to adopt smart decision-making methods and tools for big data and be responsive and adaptive to the current dynamic market, fluctuating demands, and continuous technology development. Recent technological developments in the manufacturing domain are forcing the arrangement of CPS, where the information is gathered from all related sources. The information is carefully monitored and coordinated between the physical factory networked machines on the manufacturing floor and the cyber computational areas. Therefore, to make these networked machines perform more robustly, efficiently, and collaboratively, this information needs to be stored, processed, and analyzed. As a result, the manufacturing industry is now changing to a new paradigm, which is smart manufacturing (Industry 4.0), as seen in *Figure 28*. It started from a project for advanced manufacturing ideas in 2011 that was sponsored by the German government (Xu et al. 2018). The first industrial revolution occurred at the end of the 18th century through the implementation of mechanical production using water and wind power. The second industrial revolution was in the 20th century and involved the implementation of mass production using electrical energy. The third industrial revolution was at the beginning of the 1970s and involved the use of electronics and information technology. Nowadays, the new paradigm of the fourth industrial revolution based on cyber-physical systems (CPSs) (Lu 2017, Lu 2017) will have different sorts of complexity, namely, the complexity of structure, the complexity of data, the complexity of products, and complexity through interaction. The new revolution involves using new technologies, such as cloud computing, CPS, big data, and the internet of things (IoT), Some of these technologies and processes are already being implemented in domains other than manufacturing and reaching high degrees of maturity, like cloud computing in the domain of computer technologies. These new technologies are not yet highly implemented in the manufacturing domain. The main challenge in applying these tools in the manufacturing domain is the high cost of investing in new equipment while removing outdated ones.

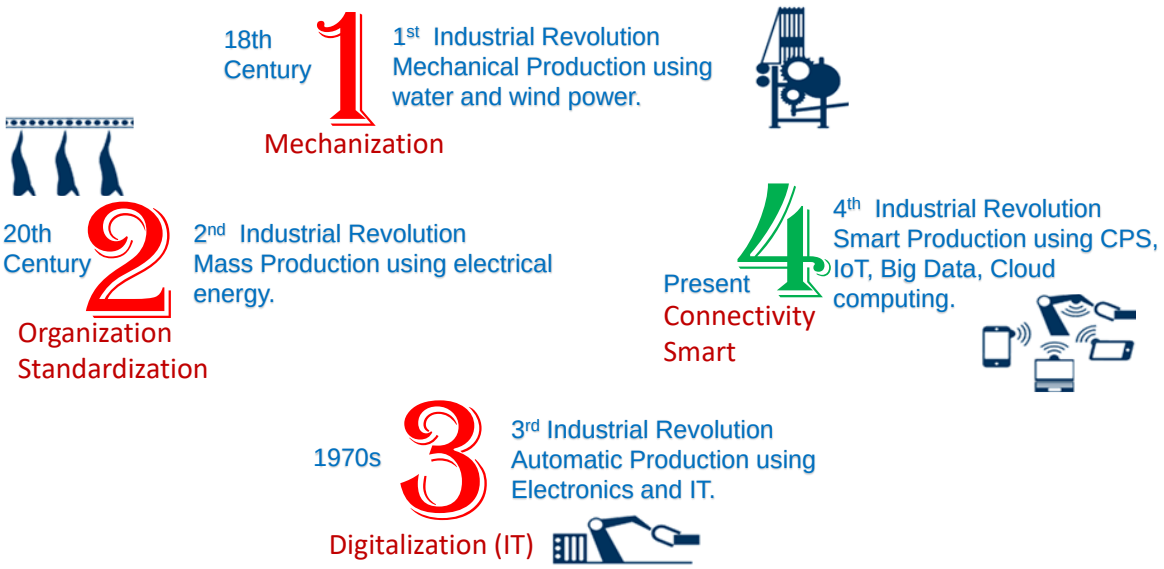

Figure 28 History of the four industrial revolutions.

4.2.4.1 Cyber-Physical Systems (CPS)

One of the main challenges faced in Industry 4.0 is CPS (Chen 2017), which defined by the U.S. National Science Foundation (NSF) as the tight conjoining and coordination between computational and physical resources (NSF 2018). (Lee et al. 2014) investigated the challenges that faced CPS and found that the main challenge is the fact that the generated data from different sources have orientations. Current-day abstraction systems do not work when it comes to the calls and thread methods. Other challenges from the system design and architecture perspective, specifically computer networking, make current manufacturing systems incompatible with CPSs (Lee 2008). (Lv et al. 2019) applied a CPS model on an automated electric vehicle using a machine-learning approach. The method involved optimizing various CPS parameters to improve different performance objectives, such as dynamic vehicle performance, drivability, and energy, along with different driving styles. The results validate the feasibility and effectiveness of the proposed CPS-based method for three driving styles which are aggressive, moderate, and conservative (Lv et al. 2019).

4.2.4.2 Internet of Things (IoT)

Within the last decade, with the new technological enhancements in the domains of communication and sensors, there is much effort to be done in the fields of CPS and the IoT (Sun 2012, Li et al. 2018). A simple definition of the IoT is that it is a system in which the objects within the physical space are connected to the internet through wired or wireless network connections through the use of the sensors attached to these objects (Babiceanu et al. 2016). It highly expected that everything around us, like devices, mobile systems, and objects, will be connected through the IoT, which will transform our big

world into one big network. This emerging technology will enable not only human but also the automated decision makers to deal and communicate with the physical equipment with the capabilities of collecting, storing, sending, and receiving generated data (Mashal et al. 2015).

4.2.4.3 Cloud Computing (CC)

The cloud computing concept appeared in the late 2000s and has been highly improved in the last few years, resulting from the significant increase in the number of applications that require cloud computing technology to operate. Sharing components and services is the key to cloud computing and significantly contributes to Industry 4.0 (Moghaddam et al. 2018). The cloud computing concept is that everything, even it is processing resources, such as central processing unit (CPUs) and graphics processing unit (GPUs), can be shared as services. That is why when we talk about software in relation to cloud computing, we talk about software as a service (SaaS), and for hardware, we talk about hardware as a service (HaaS). In cloud computing, we also discuss infrastructure as a service (IaaS), platform as a service (PaaS), and so on. We can consider anything (X) as a service (XaaS) (Mikusz 2014).

4.2.5 Materials and Methods

In the current inspection method, the iFactory inspection station camera captures the positioning of each product and makes a decision about whether it is good or bad. This is done locally using the PLC programming unit. The objective of this research is to utilize Industry 4.0 capabilities by using all I4.0 enablers instead of the former method. This process is done on a big-data analytics platform without losing the concept of I4.0. This is done by capturing images of the product and transferring them to a big-data platform on a supercomputer. The image preprocessing and decision-making are done remotely on the supercomputer. Then the results are instantly displayed in front of the end user. Choosing a big-data platform has become a significant requirement due to the increase in the volume of data and the need for timely decision-making.

The determination of product quality is classified into two main classes: a normal class in which the image of the product does not have any evidence of a defect, and a bad class in which the image shows some defects in the product. A CNN based on Tensorflow is utilized as an efficient classification tool to make such decisions. Moreover, due to the fine details and similarities in the dataset, transfer learning is used based on the Resnet-50 pre-trained model by uploading its weight matrix to our model (kaggle 2019).

4.2.5.1 Scope and applicability to upgrading existing Manufacturing Systems

The scope of this part of the research covers manufacturing systems with low, medium, and high production volumes for product families and variants. It also covers the big data with its variety

(structured, non-structured, or even semi-structured), volume (up to terabytes and zettabytes), and velocity (even real-time data). This research is applicable to new manufacturing systems and can also be used to upgrade existing manufacturing systems that have limitations. New manufacturing systems will be prepared and equipped with the required communication hardware and software that enable entire manufacturing facilities to share their resources with the IoT within the cloud. Also, some existing manufacturing systems can easily be equipped with the proper hardware and software to have the same capabilities as the new systems. For example, it is easy to install cameras and sensors in existing manufacturing systems and connect them to the internet and share the data for decision-making purposes. This can be done because the data captured from the installed cameras and sensors is analyzed to discover, qualify, and organize related information then present it to factory workers to help them make data-driven decisions to optimize the manufacturing process. Then real-time analytic actions are taken to enhance and improve production process quality and control observations. This process is already implemented in some industry fields, such as what is implemented for EigenInovations, as shown in Figure 29 (EigenInovations 2019), but still, there is a lack of research publications in this domain.

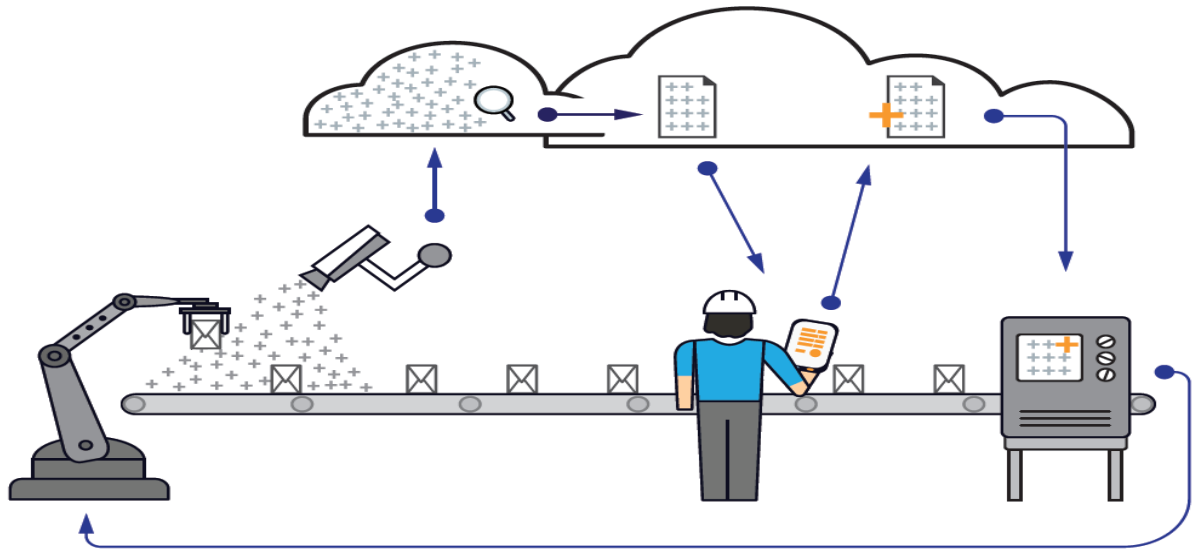

Figure 29. Upgrading existing manufacturing systems (EigenInovations 2019).

The current style of manual inspection in manufacturing systems can be time-consuming, ineffective, expensive, and sometimes, unsafe. Most of the quality checks involve visual inspections to guarantee that all parts are in the right locations, have the right colours or shapes, and are free from faults. For that reason, IBM proposed the use of artificial intelligence technology called IBM Visual Insights to quickly resolve quality problems, minimize escape rates, decrease the number of required manual inspections, improve the effectiveness of quality inspectors, and advance overall throughput across multiple industries. They indicated that the implementation of IBM Visual Insights has some benefits, such as

reducing the cost of quality inspection, reliability improvements for the inspection process, reductions in inspection time, and root-cause analysis and problem resolution improvements (IBM 2019).

Modern manufacturing systems are becoming increasingly interconnected and complex, and they are generating new challenges that automation can address, such as quality checks at machines and inspection stations. Similar to other businesses, Intel's long term automation vision for smart manufacturing is based on the realization that big-data analytics can dramatically increase efficiency, productivity, quality, and safety and decrease costs. Utilizing computing power and the IoT to have an automated control system with real-time data that is stored on off-line systems as big-data servers for future analysis and real-time decision-making by transforming data into actionable information. As proposed by Intel, real-time/non-real-time data analytics is shown in Figure 30 (Steve Chadwick 2016).

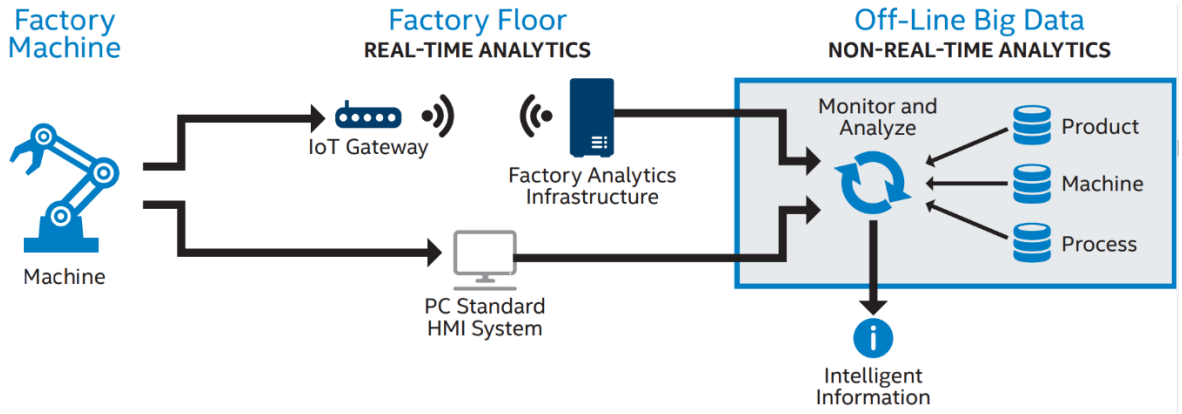

Figure 30. Real-time /Non-real time data analytics (Steve Chadwick 2016).

While real-time monitoring using data analytics for the quality rules is applied to real-time process data to determine whether it is feasible to continue production of a product, whether it requires rework, or whether it should be scrapped. Rapid responses to problems also increase uptime, accelerate output, and save money, as shown in Figure 31 (Steve Chadwick 2016).

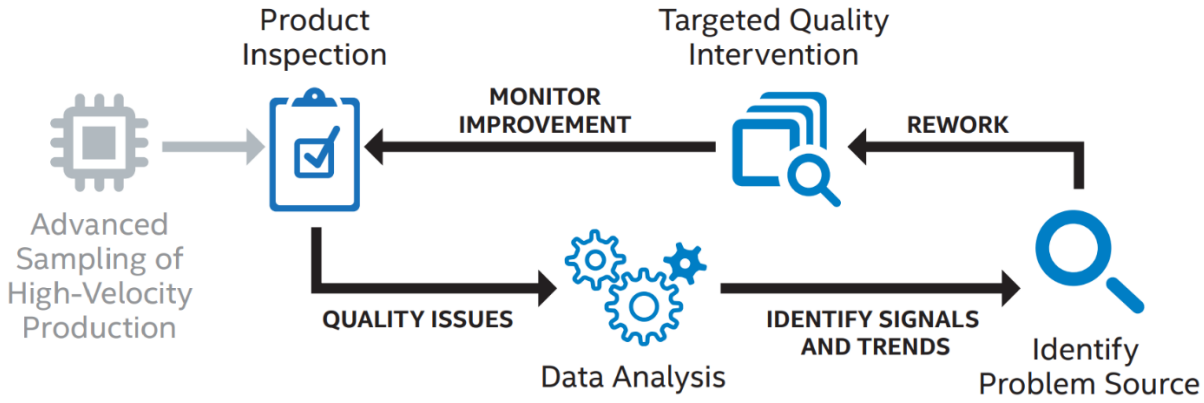

Figure 31. Real-time monitoring using data analytics (Steve Chadwick 2016).

(Moyne et al. 2017) proposed the use of big-data analytics for smart manufacturing in semiconductor industries and defined smart manufacturing as the enhancement of manufacturing operations through the integration of systems, connecting cyber and physical capabilities, and capturing the advantages of information, including big-data growth, by applying big-data analytics for improver prognostics and diagnostics. This will be done by improving current capabilities, such as fault detection, and adding new capabilities, such as predictive maintenance. The essential big-data factors are data quality and analytic techniques for achieving high-quality solutions. Due to the highly complex and dynamic equipment and processes, as well as issues in manufacturing data size and quality, it is emphasized for use in next-generation fault classification and detection and predictive maintenance.

4.2.5.2 Data Acquisition

In this phase, the camera (with a resolution of 1024 by 1024 DPI) is mounted to a base in the inspection station of the iFactory and used to capture product images. These images are in RGB (red, green, and blue) format. The number of acquired images is 20,000 to build a sufficient dataset; these images are transferred to a supercomputer and split into training, validating, and testing datasets. The training dataset images are labelled using natural language processing with a precision of 99% accuracy. Then after the datasets are processed, they are fed into the machine-learning model.

In *Figure 32*, two samples of normal and bad classes are illustrated. The left image is the normal class, and the right image is the defected one. The labelled dataset and images make the problem a supervised machine-learning classification problem. The normal class is labelled as good and represented by 1, and the defected class is labelled as bad and represented as 0. The number of samples in each class is balanced and set to 10,000 images.

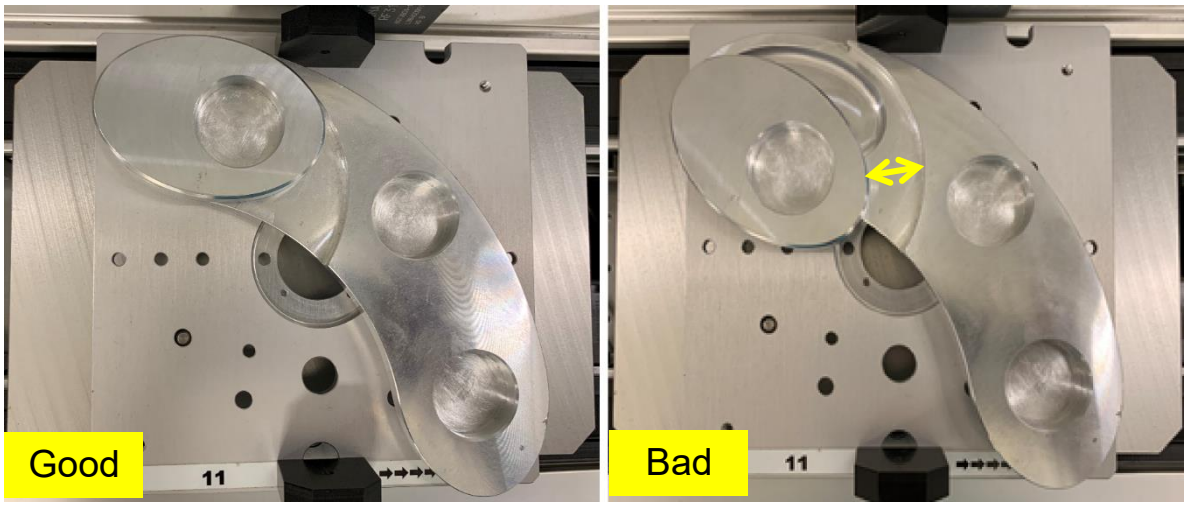

Figure 32. Normal class, good image (left), and defected class, bad image (right).

4.2.5.3 Pre-processing

The preprocessing phase prepares the images to fit within the machine-learning model to retrieve the best features quality. In this phase, the images are normalized by scaling them down to 512 by 512 pixels. Then all image channel values are normalized to be between 0 and 1. This is done by dividing all pixels' values by the highest pixel value, which is 255. The reason for this is to remove the variation in channel values and accelerate the convergence of the machine-learning model. In the second phase, image augmentation is performed to increase the number of images and minimize the probability of overfitting. The augmentation is done by randomly and horizontally flipping a certain proportion of images and rotating them by 45 degrees. The features acquired for each batch are captured by cropping the image from the centre and rescaling it to 224 by 224 pixels. The labels of images are either normal or bad and represented in one hot encoding vector by 1 or 0, respectively.

4.2.6 Model Architecture - Methodologies and Model Design

The model consists of three phases: the first phase is to read the dataset that is stored on a supercomputer. In the second phase, pre-trained model weights from Resent-50 are utilized as initial weights in our model. Resnet-50 consists of 227 layers, which makes the machine-learning model deep. Our model is trained from end-to-end layers (without freezing layers) to capture the fine details of each class and extract discriminative features between both of them. Each layer in the Resnet-50 consists of a convolution layer, a max-pooling layer, a ReLU activation function, as given in Equation 8, and a dropout optimization technique. The first layer of the network is replaced with an input layer, and the last layer is chopped and replaced with two fully-connected class layers. The SoftMax nonlinearity activation function given in Equation 9 is applied with a binary cross entropy loss function. The state-of-the-art method called ADAM (a method for stochastic optimization method), which is stochastic gradient descent, is used to optimize the loss function. Figure 33 shows a sample of layers of CNN.

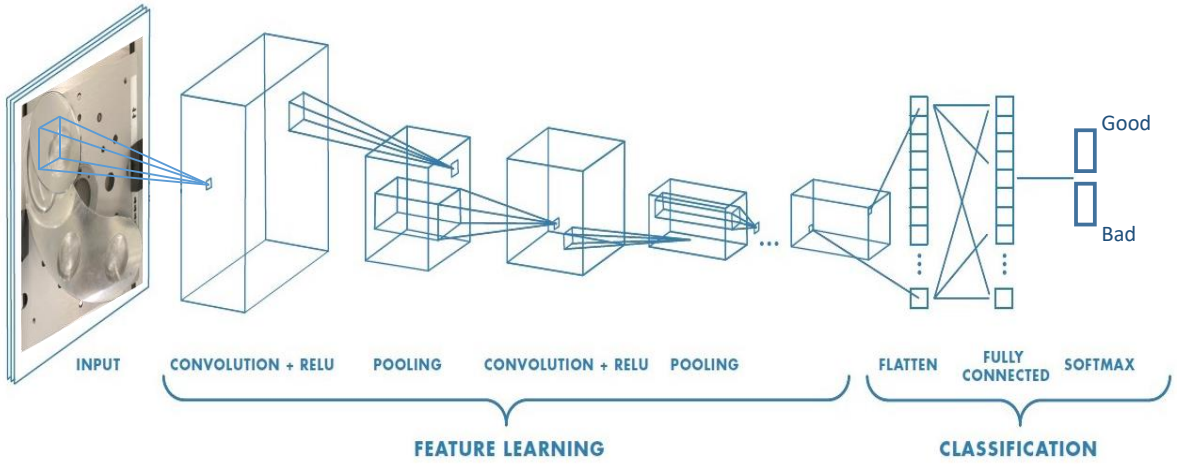

Figure 33. Structure of convolutional neural networks (CNN) (FreeCodeCamp, 2019).

The reason for using a CNN is its efficiency in feature extraction by multiplying images with a certain kernel or filter. The main unit of layers consists of a convolution layer that extracts the feature by multiplying the image with a certain size of the kernel. Convolution preserves the spatial relationship between pixels by learning image features using sliding small squares of input data. The kernel filter slides from the left of the image to the right of the image with one or two overlapping pixels or stride. In this model, the kernel size was a 3 by 3 matrix, which is also called a filter. Changing the values of the filter matrix will produce different feature maps for the same input image. Figure 34 shows the extracted features when a canny filter is used.

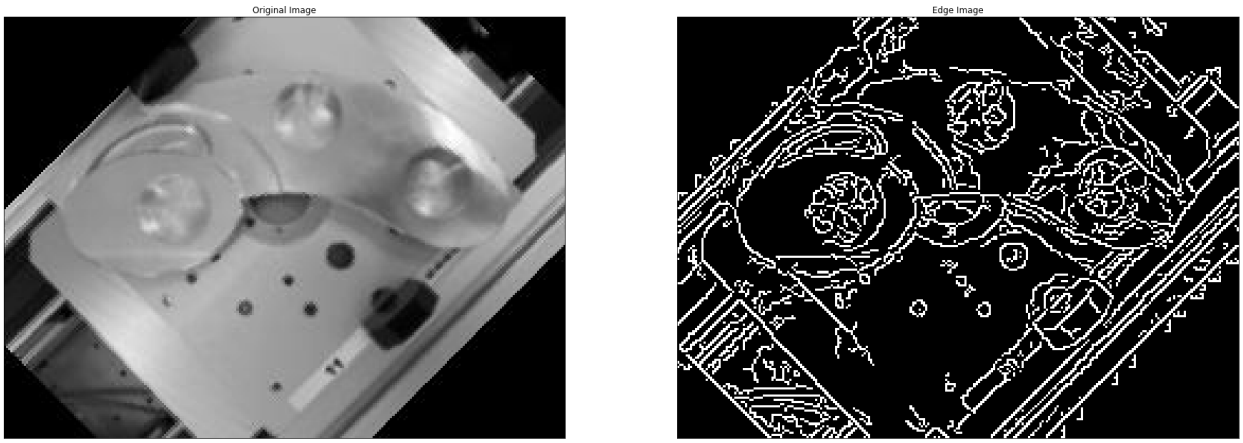

Figure 34. Input image with its different feature maps.

The second layer is the pooling layer, and the main purpose of this layer is to train the model about the object regardless of its place in the images. For example, if a max-pooling layer is used with 2 by 2 for every four cells, then the maximum numerical value is computed and inserted into the pooled feature map. Then the activation functions, such as ReLU and SOFTMAX, shrink the output number that is produced from the former layers. These numbers form the weight matrix of the neural network. The goal is to find the best weight matrix that minimizes the loss function, which is binary cross entropy loss function which is given in (Equation 7):

$$CE = -\sum_{i=1}^{C'=2} t_i \, log(s_i) = -t_1 \, log(s_1) + (1 - t_1) \, log(1 - s_1) \quad (7)$$

The loss function measures the difference between the input and the predicted output. The used activation function is Rectified Linear Unit ReLU. As shown in Figure 35, ReLU is an element-wise operation (applied per pixel) and replaces all negative pixel values in the feature map with zero. The purpose of ReLU is to introduce nonlinearity to our ConvNet because most of the real-world data we would want our ConvNet to learn would be non-linear. Convolution is a linear operation–element-wise

matrix multiplication and addition function, so we account for nonlinearity by introducing a non-linear function like ReLU (Hahnloser et al. 2000).

$$output = max(\,0\,, input\,) \qquad (8)$$

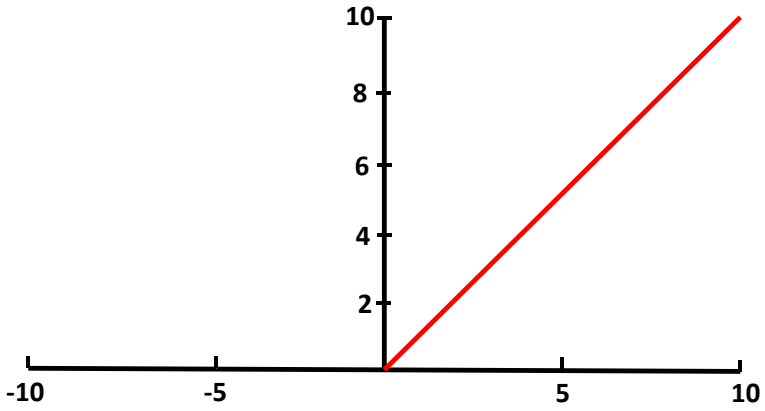

Figure 35. Rectified Linear Unit RELU.

The weight matrix keeps updating using forward propagation and backward proportion. The last fully connected layer with two classes is used with softmax (Boltzmann 1868) as shown by the following equation.

$$\alpha(x_j) = \frac{e^{(x_j)}}{\sum_i e^{(x_i)}} \qquad (9)$$

where α is the output of the activation function, x_i is the weight matrix of layer j, and x_j is the output of layer i. as shown in Figure 36 below.

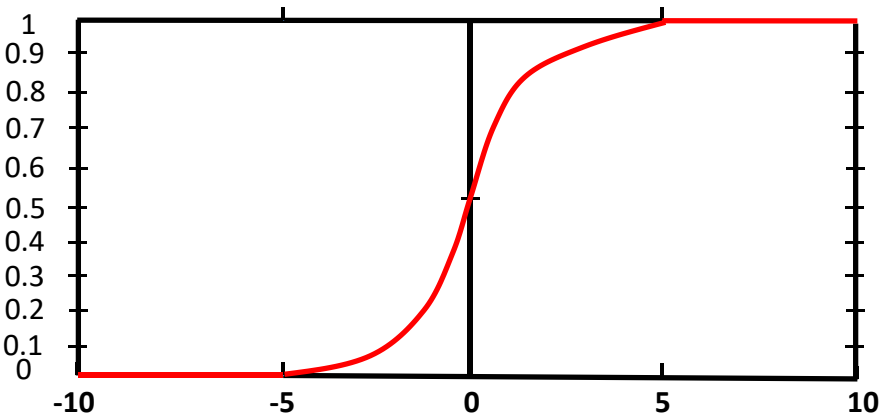

Figure 36. Softmax activation function.

These two classes (bad and good) are denoted in order as 0 and 1. The number of layers used is 227 because the model was built based on the idea of transfer learning using the Resnet-50 pre-trained

model. However, Resnet -50 was trained from scratch, and all weight matrix values were discarded for the sake of improved learning (kaggle 2019).

4.2.7 Tuning the Model Hyper parameters

The hyper parameters are tuned in this approach by observing the accuracy of performance. The number of epochs is set to stop early with the max accuracy score set to 99%, and the model training is done after 10 epochs with the targeted accuracy. Differing batch sizes of 20, 40, and 60 images per batch were used. The dropout value was set to 0.2 to prevent overfitting and improve accuracy. The learning rate is tuned using a learning rate scheduler with a base learning rate of 1e-7 worm up to 5 epochs (0.001) then cool down to a minimum learning rate of 1.E-15. However, the best learning rate for which the best accuracy is achieved was 0.001. The optimizer used to minimize the loss function was ADAM.

4.2.8 Results

A supercomputer was utilized to test this application using Apache Spark, which is an open source general-purpose cluster computing framework. Spark also provides an interface for programming clusters with data parallelism and fault tolerance. The model was executed on a Compute Canada supercomputer in a cluster with four nodes known as Cedar. Each node has 16 cores and 100 GB of memory per CPU. The data was split into training (70%), validation (20%), and testing (10%) datasets. The model was tested with different batch sizes, such as 20, 40, and 60 images. The results are given in Table 4, which shows the performance measurements, such as the accuracy and loss of the classification. Accuracy is known as the percentage of the number of correctly classified images divided by the total number of images, while loss is defined as the least square function that takes the square root of all the added-up squared differences between the prediction value and the actual value for each sample. All the experiments were conducted using big-data processing tools and parallel computing using Apache Spark with Python. By using a different number of nodes (to measure the scalability of the system), more data will be processed in less time than ever.

Table 4. Results (Accuracy and loss) With different batch sizes, Epochs, and Drop out.

Batch size	**Dropout**	**0.1**			**0.2**		
	# of Epochs	**5**	**10**	**15**	**5**	**10**	**15**
20	**Accuracy**	55.87%	76.85%	95.84%	80.68%	96.78%	96.75%
	loss	69.35%	51.82%	0.99%	43.18%	2.17%	5.61%
40	**Accuracy**	51.27%	72.56%	93.79%	73.23%	92.45%	94.99%
	loss	69.25%	51.61%	9.09%	40.41%	10.99%	0.25%
60	**Accuracy**	48.16%	60.96%	88.91%	55.78%	62.45%	89.41%
	loss	68.95%	65.41%	14.96%	69.32%	60.87%	21.63%

4.2.8.1 Performance

The data was split into training (70%), validation (20%), and testing (10%) datasets. The performance was measured based on how many images can be processed per epoch per time. It was noticed that with increases in batch size, accuracy and training time decreased. However, using a small batch size takes a long time but results in high accuracy. We used a batch size of 60 with 10 epochs, and each epoch lasted approximately 7 minutes, but 62.45% accuracy was achieved. However, when we used a batch size of 20 with the same number of epochs, each epoch lasted 20 minutes, and 96.78% accuracy was achieved.

4.2.8.2 Accuracy with Different Batch Size and Dropout

In this experiment, the accuracy was measured based on the batch size with a dropout of 0.1. It is noticed that with the increase in batch size, the training time decreases and accuracy increases. The number of epochs was set to 5, 10, and 15. The best results were achieved for 12 epochs with a batch size of 20 and 95.84% accuracy. The best learning rate was 0.001. Figure 37 depicts the results based on the accuracy, batch size, and a number of epochs when dropout was 0.1.

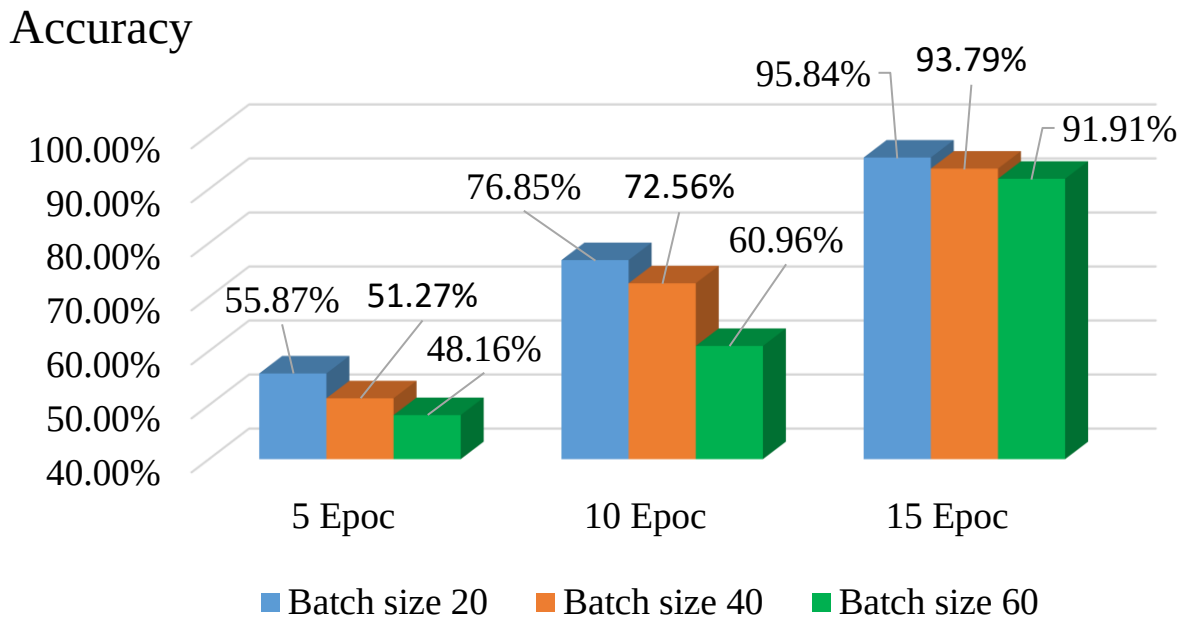

Figure 37. Results based on the accuracy, batch size, and the number of epochs with dropout 0.1.

In this experiment, accuracy is measured based on batch size with differ dropout of 0.2. It is noticed that with increases in batch size, the training time decreases but accuracy increases. The number of epochs was set to 5, 10, and 15. The best results were achieved when the number of epochs was 10 with a batch size of 20, 96.78% accuracy, and the same learning rate as in the above experiment. Figure 38 shows the results based on the accuracy, batch size, and a number of epochs, and drop out was 0.2.

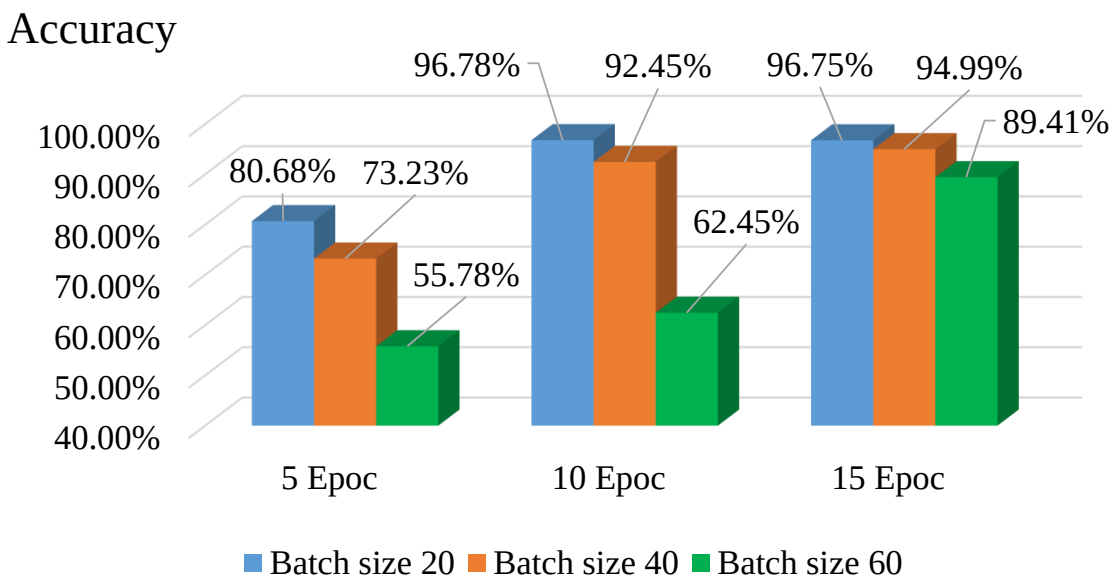

Figure 38. Results based on the accuracy, batch size, and the # of epochs with dropout 0.2.

4.2.9 Conclusion

Industry 4.0 is a modern global trend is sustainable manufacturing systems. The main contribution of this research is in implementing a key enabler for Industry 4.0, which is big data analytics using AI. The proposed method can be used in optimizing manufacturing processes among a wide variety of industries. The successful implementation of this research in manufacturing facilities will lead to not only the realization of cost savings in these facilities but also the achievement of feasible sustainable systems.

The pipeline alike method starts with the training of the model on the acquired data of both good and defected product images from the production line. The data is preprocessed and normalized to enhance overall machine-learning performance. The trained data is located on a supercomputer and synchronized with the production line in real-time. The production line takes a picture of each product item and sends it to the server to be classified using the trained machine-learning model. If an image is classified as defected, then the server will instruct the production line to take the proper action, such as sending the item to an alternate station. The integrated models have a high-performance level, with up to 97% of accuracy for detecting defected parts. The model is a real-time decision-making machine that is connected to a supercomputer and implemented all the missing I4.0 enablers in the iFactory system.

In order to classify a product as defeacted or normal, I used a novel classification procedure based on nueral network methodology to achieve a real-time decision making in manufacturing system. In addition, I also collected the data from the production line and trained the model through cloud servers before taking the decision. This is a novel contribution because it is the first transparent research attempt to automate decision-making in a manufacturing setting.

CHAPTER 5. HEALTH INFORMATICS APPLICATION

5.1 Overview

Recently, the advancement in genetic sequencing and gene expressions tools boost the study of the human genome. Human genes play roles in daily activities and can be a cause or a result of any disease. Breast cancer is not an exception, and the study of the human gene in relation to it assist in the early detection, diagnose, and the treatment of the disease. Breast cancer was classified into five molecular subtypes by the genetic analysis (Perou et al. 2000) (Prat et al. 2011), then, later on, (Curtis et al. 2012) classified breast cancer into 10-molecular subtypes based on the genetic activities. The vast amount of genetic data requires different pre-processing and analysis methods, the number of human genes is estimated to be around 25,000 coding genes and about 22,000 for non-coding genes, those genes have a different way of transcriptions which is known as transcripts, where the total number of known transcripts is 198,002 (Biostars 2019). The relationships among those genes themselves and the other molecules, proteins, motives, and conditions complicate the computational model for studying any genes. The newer generated genomic data over the public studies and dataset on the internet makes this problem to be considered as a Big data problem.

The literature shows many integrative machine learning models for breast cancer treatment, subtypes, and survival. However, this is the first model that analyses the genes in treatment-survival data. This case study proposes a machine learning model that learns from the big data of gene expressions and guide the treatment of the patient throughout five years' survival interval.

The main contribution of this chapter is the novel algorithm that is followed in order to classify a multi-class outcome based on a large number of features.

5.2 Case Study 2

A machine learning approach for identifying gene biomarkers guiding the treatment of breast cancer. (Tabl et al. 2019)

5.2.1 Abstract—Objective

Studying breast cancer survivability among different patients who received various treatment therapies may help us understand the relationship between the survivability and treatment of patients based on the gene expression. In this work, we present a classification system that predicts whether a given breast cancer patient who underwent through hormone therapy, radiotherapy, or surgery will survive beyond five years after treatment. Our classifier is a tree-based hierarchical approach that groups breast cancer patients based on survivability classes. Each node in the tree is associated with treatment therapy and a subset of genes that can best predict whether a given patient will survive for more than 5 years after that particular treatment. We applied our tree-based method to a gene expression dataset about 347 treated breast cancer patients and identified potential subsets of biomarkers that can predict survivability with high accuracy levels, ranging from 80.9% to 100%. We investigated the roles of many biomarkers through a literature review and found that certain biomarkers are strongly related to breast cancer survivability.

5.2.2 Background

Despite the fast increase in the breast cancer incidence rate, the survival rates have also increased due to improvements in the treatments because of new technologies (Miller et al. 2016). Breast cancer, however, is still one of the leading causes of cancer-related death among women worldwide. The survival rates vary among the various treatment therapies that are currently used, which include surgery, chemotherapy, hormone therapy, and radiotherapy. Nevertheless, each patient's response to a specific treatment varies based on some factors that are being investigated (Miller et al. 2016).

Traditional laboratory techniques like CAT scans and magnetic resonance imaging (MRI) have been proven to be useful. However, they offer very little information about the mechanism of cancer progression. On the contrary, advances in DNA microarray technology have provided high throughput samples of gene expression. Machine learning approaches have been utilized to detect breast cancer treatment or survivals (Tang et al. 2017) (Zeng et al. 2018) (Mangasarian et al. 2000) (Cardoso et al. 2016) (Abou Tabl et al. 2017). many researchers have used DNA microarray technology to study breast cancer survivability (Mangasarian et al. 2000) (Cardoso et al. 2016) (Abou Tabl et al. 2017). Analyzing gene expression among breast cancer patients who undergo varying treatment types deepens the current

understanding of the disease's progression and prognosis. Many features complicate the computational model; the number of features is usually significantly larger than the number of samples, which is known as the curse of dimensionality problem, in which standard classifiers overfit the data, and hence, perform poorly. Therefore, feature selection techniques are proven to alleviate the curse of dimensionality by removing irrelevant and/or redundant features.

Zou et al. proposed a maximum Relevance maximum distance feature selection approach mRMD. The method uses Pearson's correlation coefficient to measure the Relevance between the subgroup of features and the class. The selection criteria balance the accuracy with stability when selecting the features. The authors compared the dimensionality reduction method with both filter and wrapper feature selection types, and the results show that mRMD outperformed different features selection method of each type (Zou et al. 2016). We compared mRMD with mRMR on the wrapper phase of feature selection, while the accuracy of random forest on the selected features of each method was very close, mRMR overall selected less number of potential biomarkers with 47 genes compared to 60 genes of the mRMD, Hence, we utilized mRMR in this model to obtain a handful smaller size of potential biomarkers for further analysis.

Tang et al. predicted a tumour location in breast tissue based on feature selection method where the features are RNA-Seq and miRNA data (Tang et al. 2017), they enhanced the prediction of the standard classifiers to be around 93% in average. While Zeng et al. investigated a potential miRNA biomarker for breast neoplasm with around 80% accuracy (Zeng et al. 2018). Mangasarian et al. utilized a linear support vector machine (SVM) to extract 6 out of 31 clinical features. Their dataset contains samples from 253 breast cancer patients. The model involved classifying the samples into two groups: (1) the node-positive group in which the patients have some metastasized lymph nodes, and (2) the node-negative group for patients with no metastasized lymph nodes. Those six features were then used in a Gaussian SVM classifier to classify patients into three prognostic groups: negative, middle, or positive. The researchers found that patients in the negative group had the highest survivability. Most of these patients had received chemotherapy treatment (Mangasarian et al. 2000).

Using samples from patients with high-risk clinical features in the early stages of breast cancer, Cardoso et al. proposed the use of a statistical model to determine the necessity of chemotherapy treatment based on clinical data (Cardoso et al. 2016). In one of our earlier works, we built a prediction model based on various treatments without defining the period of survivability (Abou Tabl et al. 2017); that is, given a training dataset consisting of gene expression data of BC patients who survived or died after receiving a treatment therapy, we built a classification model that is used to predict whether a new patient will survive or die. In another work, we have implemented an unsupervised learning approach to find the separation between the treatment-survival classes (Tabl et al. 2018), the model is grouping different classes together in building the tree model while defining the border between the different groups of

classes. Paredes-Aracil et al. built a scoring prediction system for 5 and 10 years of survivability periods for different BC subtypes. The cohort of their study includes 287 patients from a Spanish region. The patients have received different therapies with sometimes mixed of them (Paredes-Aracil et al. 2017), which makes it difficult to relate the genomic activities to a specific therapy during the survival prediction.

In this present work, we extended an earlier work (Tabl et al. 2018) to predict which BC patients will survive beyond five years after undergoing a given treatment therapy. The classifier is built on top of a feature selection model that identifies the genes that can be used to best distinguish among patients in differing survivability classes. To the best of our knowledge, this work is the first supervised machine learning method is built to predict the BC five survivability for each therapy.

5.2.3 Materials and Methods

We used a publicly-accessible dataset that contains samples for 2,433 breast cancer patients (Pereira et al. 2016) (Curtis et al. 2012). The gene expression profiles were totally processed and normalized (Curtis et al. 2012). After studying the given data and selecting only patients who have received one type of treatment, a set of six classes were identified as the base of this work. These classes are the mixture of each treatment: surgery (S), hormone therapy (H), and radiotherapy (D) with a patient status as living (L) or deceased (D). The numbers of samples (patients) for each class in the proposed model are shown in Table 5. Data from a total of 347 patients were included in this work. To avoid overfitting, we performed the filter feature selection first for each class, before running the wrapper feature selection or even the classification model on all the samples from all classes. The number of genes after the two feature selection steps for each class are reported in Table 5.

Table 5. list of classes with the number of samples in each class, with the number of genes for each class after filter feature selections.

Class	Number of samples	Number of genes after filter feature selection	Number of genes after wrapper feature selection
Living and Radio (LR)	132	1771	8
Living and Surgery (LS)	130		
Living and Hormone (LH)	20	80	9
Deceased and Hormone (DH)	6	20	10
Deceased and Radio (DR)	19	227	14
Deceased and Surgery (DS)	40	197	6
Total	**347**	**2295**	**47**

Based on the available data, only three treatment therapies are covered in this study: surgery, hormone therapy, and radiotherapy. Our model uses hierarchical classifiers to classify one-versus-the-rest classes. The classes are imbalanced. Hence, standard classification methods will yield poor performance results. The pipeline starts with feature selection methods like Chi-square (Mantel 1963), and information gain (IG) that are applied to limit the number of significant features (genes). A wrapper method is also utilized to obtain the subset of genes that best represents the model by using minimum redundancy maximum relevance (mRMR) (Peng et al. 2005), as a feature selection method. This step is followed by several class balancing techniques, such as the synthetic minority over-sampling method (SMOTE), resampling, and cost-sensitive to balance the number of classes before applying different types of classifiers, such as naive Bayes (Domingos et al. 1997) and decision tree (random forest) (Breiman 2001). Finally, a small number of biomarker genes is recognized for predicting proper treatment therapy for the patient. To the best of our knowledge, this work is the first prediction model that is built on the combination of the treatment and survivability of the patient as a class.

The patient class distribution for the studied model is shown in Figure 39, which shows the percentages of samples within each class. It is clear that there are differences between classes that require class imbalance handling techniques to achieve fair classification.

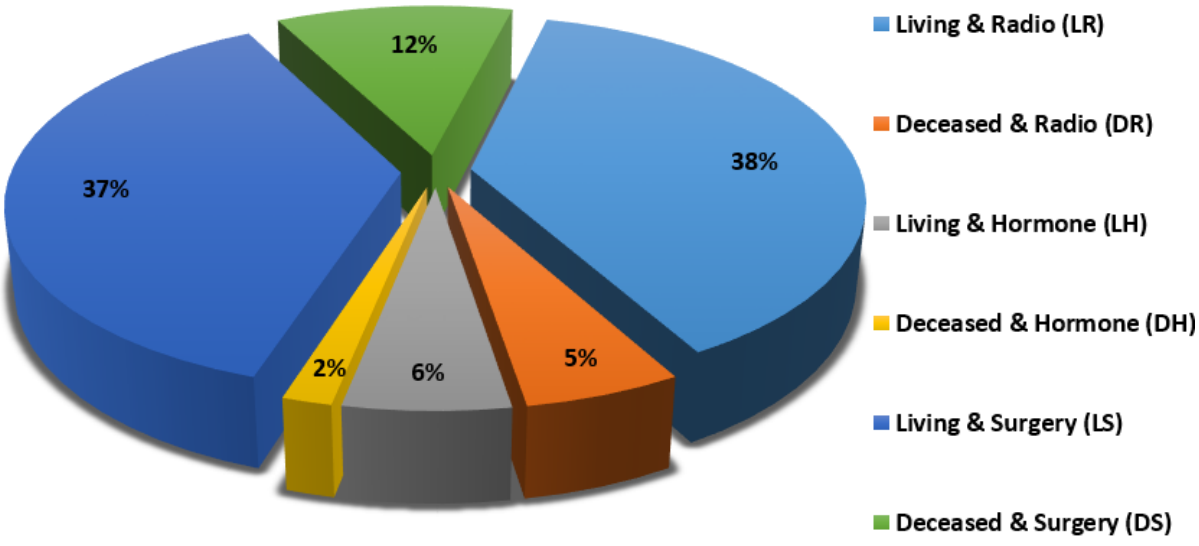

Figure 39. Patient class distribution.

5.2.3.1 Class Imbalance

This model uses a one-versus-rest scheme to tackle the multi-class problem, which leads to an imbalanced class dataset at each node of the classification model because of the differences in sample sizes. Standard classifiers are bias to the class with higher number of samples (the major class) in a

way usually they tend to select the major class class over the minor class (Elkarami et al. 2016) . Differences in class priorities due to the sample is a common problem in machine learning, nonetheless, the dataset in this work is a non-blanced dataset. Therefore, we applied the following techniques to handle this issue:

5.2.3.1.1 Over-sampling with synthetic data

Oversampling the minority class by using synthetic data generators. Several algorithms are used to achieve this. We used one of the most popular ones, SMOTE (Chawla et al. 2002).

5.2.3.1.2 Using a cost-sensitive classifier

Using penalizing models that apply additional weight to the minority class to achieve class balancing. This, in turn, biases the model to pay more attention to the minority class than others. The algorithm utilized in this work is called Cost-Sensitive Classifier in the Weka machine learning tool using a penalty matrix to overcome the imbalance (Núñez 1988).

5.2.3.1.3 Resampling

Replicating the dataset can be using one of two methods: (1) adding copies of the data instances to the minority class, which is called over-sampling (2) deleting some instances of the majority class, which is called under-sampling. We used the over-sampling technique (Gross 1980).

5.2.3.2 Feature Selection

The gene expression dataset contains 24,368 genes for each of the 347 samples. Because of the curse of dimensionality, it is difficult to classify the dataset in its current form. Thus, engaging in feature selection is essential to narrow down the number of genes to a handful at each node. Chi-square and Info-Gain are applied to select the best information gain of the selected genes, this step (Which is usually called filter feature selection) will drop down the number of genes to a couple of hundreds based on the correlation between each class and the gene expressions based on the default correlation threshold in WEKA. After that, mRMR is applied to identify the best subset of significant genes. mRMR (Which is usually considered as a wrapper feature selection) is an algorithm that is commonly used in greedy searches to identify the characteristics of features and correctly narrow down their relevance.

In the trial to find the best feature selection wrapper method to select handful gene biomarker for each class, we applied both mRMD 2.0 and mRMR on the filtered genes on each class. mRMD outperformed mRMR fourth and the fifth node as seen in Table 6, while mRMR performed better in the second and third. Both classifiers had 100% of accuracy in the first node, but the lower number of selected genes in that node using mRMR made it more efficient.

Table 6. illustrates the results of using mRMD 2.0 vs mRMR on each node then applying random forest classifier on each node.

Node	mRMD		mRMR	
	# of Biomarkers	**Accuracy**	**# of Biomarkers**	**Accuracy**
DH VS Rest	20	100.00%	10	**100.00%**
DR VS Rest	13	99.47%	14	**100.00%**
LH VS Rest	4	98.25%	9	**100.00%**
DS VS Rest	13	98.69%	6	**97.90%**
LR VS LS	10	81.29%	8	**80.90%**
Total # of Biomarkers	**60**		**47**	

5.2.3.3 Multi-class Classification Model

We applied a multi-class approach, the one-versus-rest technique. This approach involves classifying one class against the remaining classes and then removing that class from the dataset. Afterward, we selected another class to classify it against the rest, and so on. Using this greedy method to find the starting node, the method involves classifying all possible combinations, such as DH, against the rest, then DR against the rest, and so on for all six classes. Afterward, the best starting node is selected as the root node for the classification tree based on the best performance.

Several classifiers were tested to achieve these results, including random forest, support vector machine (SVM), and naive Bayes, random forest outperformed the others and has shown a better classification power for the hierarchical model. Therefore, we used it in all nodes. The classification model was built using 10-fold cross-validation. The data is divided into 10 equal folds of samples, then the learning method will loop 10 times, at each time, it will learn from 90 folds, and test on the remaining (left out) fold. At each time in the loop, it will take out a unique fold that has not been shown up in the previous loop steps as a left out. The 10-fold modules will increase the learning samples to 90% of the samples, while it will test on 100% of the samples. The sample will be classified around 9 times; the class is voted more will be considered as the predicted class. The accuracy and other performance measurements are calculated based on the testing folds; therefore, the accuracy here is a testing accuracy.

5.2.4 Results and Discussion

The developed multi-class model also shows the final results for each node and the performance measures that were considered, such as accuracy, sensitivity, F1-measure, and specificity. Moreover, it also shows the number of the correctly- and incorrectly-classified instances in each node.

In Figure 40, the root node is DH against the rest that gives 100% accuracy. The second node is obtained after removing the DH instances from the dataset and then classifying each class against the rest. The best outcome was DR, which had an accuracy level of 100%. We repeated the same technique for the third node, finishing with LH with an accuracy of 100%. Then DS in the fourth node with an accuracy of 97.9%, sensitivity is 96.9%, and specificity is 100% because all the DS samples were correctly classified. In the fifth node, which is the final one, we have LR and LS. The accuracy drops down to 80.9% because it is difficult to distinguish between the living samples in both.

Our method was used to identify the 47 gene biomarkers that are listed in Table 7. Functional validation was conducted, and biological Analysis was provided for some genes by studying the information provided in the literature. The genes marked as blue are those that were considered for further biological relevance (see the discussion in the next section).

Table 7. Gene biomarkers for each class versus the rest at each node.

	DH	**DR**	**LH**	**DS**	**LR and LS**
Genes	AKIP1	ASXL1	DA874553	ICOSLG	C14orf166
	FGF16	WIPI2	AKT1S1	SAR1A	ZFP91
	AA884297	ASAP1	CPPED1	PRPS1	BU753119
	CDC42BPG	ZNF121	BLP	FBRSL1	ARPC3
	UPF3B	METTL2A	ARFGAP2	INPP5F	OSTC
	FAM114A1	FAM170B	VAMP4	SFMBT2	AI376590
	OR2G6	BG944228	CT47A1		OR2B3
	ANKLE1	PDCD7	CLASRP		DSCAM
	MGA	ATL1	CD36		
	C14orf145	TRPC5			
		FOSB			
		AL71228			
		BF594823			
		FBXO41			

At each node, we tried different standard classifiers to select the classifier with the best accuracy at that node, as seen in Table 8; random forest outperformed the other classifiers in all nodes. The accuracy at the difficult node 5 still down compared to the other nodes. However, we can see a significant improvement in this node as it is 80.9% comparing to the second best 77.1% accuracy using SVM with a linear kernel. In node 4, where the accuracy is 97.9% for the random forest, the other classifiers performed with very low 79.06% accuracy for the second best, which is SVM with radial basis function kernel. Bayesian classifier had the second best performance in the first, second, and third nodes with 99.47%, 96.3%, 92.4% accuracies in order. SVM with polynomial degree 3 kernel had an average performance in all nodes compared to the other classifiers.

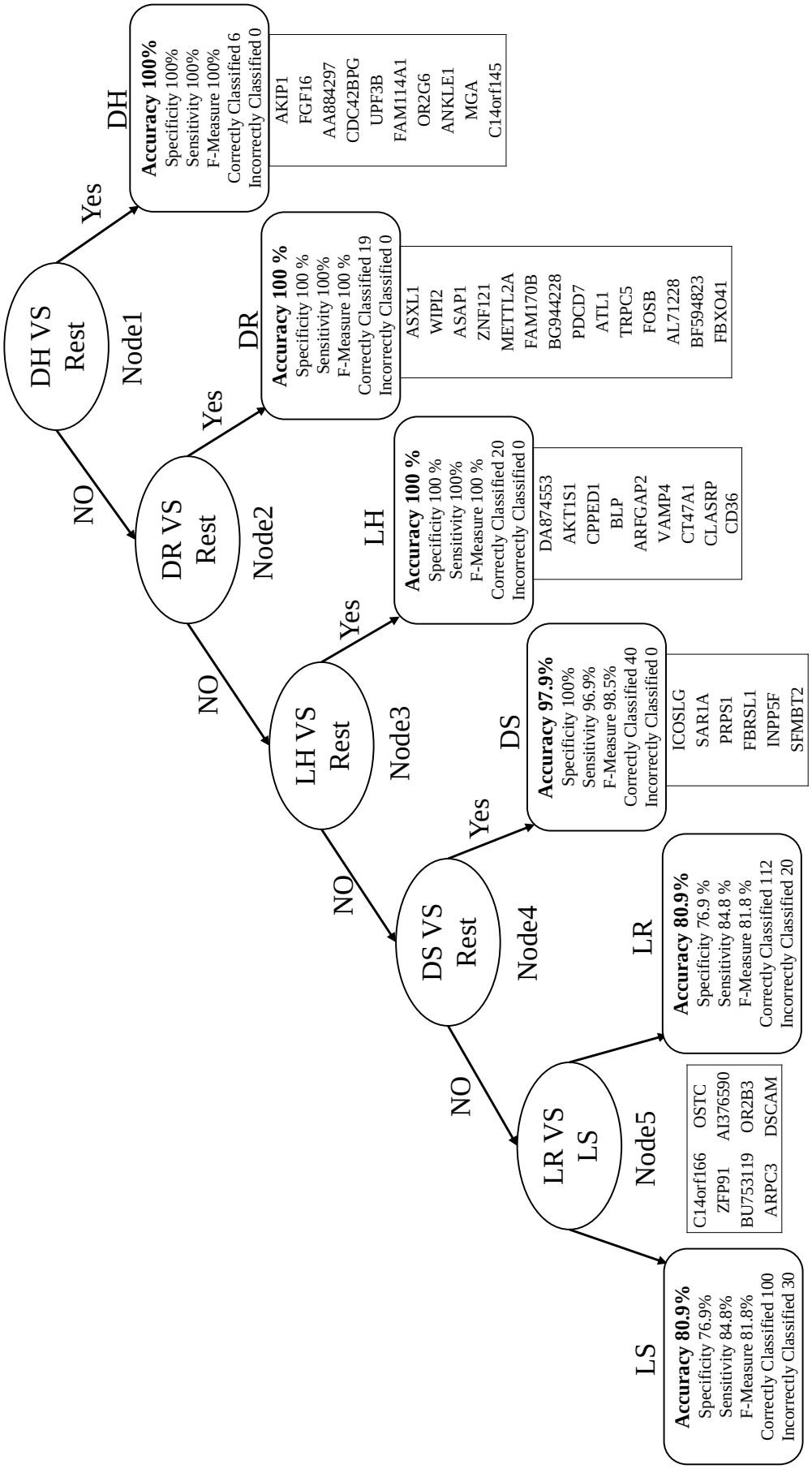

Figure 40. Multi-Class classification model with performance measures.

Table 8. Comparison of the standard classifiers at each node of the proposed model.

Node	SVM Linear	SVM Polynomial	SVM RBF	Bayesian Naive Bayes	Random Forest
DH VS Rest	98.41%	98.68%	97.35%	99.47%	**100%**
DR VS Rest	94.46%	95.78%	91.56%	96.3%	**100%**
LH VS Rest	89.47%	92.4%	88.3%	92.4%	**100%**
DS VS Rest	75.65%	77.23%	79.06%	75.92%	**97.9%**
LR VS LS	77.1%	74.81%	76.72%	76.34%	**80.9%**

5.2.5 Biological Insight

A combination of gene regulation analysis and biological analysis have been done to validate some of the biomarker genes. Biological validation was carried out using relevant literature (Katoh et al. 2014) (Kechavarzi et al. 2014) (Sabe et al. 2009) (Bamberger et al. 1999) (Allegra et al. 2012) (Caballero et al. 2014) (Qiu et al. 2015) (Nam et al. 2015) (Dombkowski et al. 2011) (Tommasi et al. 2009).

Figure 41 is a multi-dimensional representation of the plot matrix for the six biomarker genes found in Node 4 for the DS class versus the remaining ones, as an example. The figure also shows the relations among the six genes. It is clear from the class column that the samples are separable. The values in x-axis represent the gene expression values in the column side, where the y-axis represents the gene expression values at the raw side.

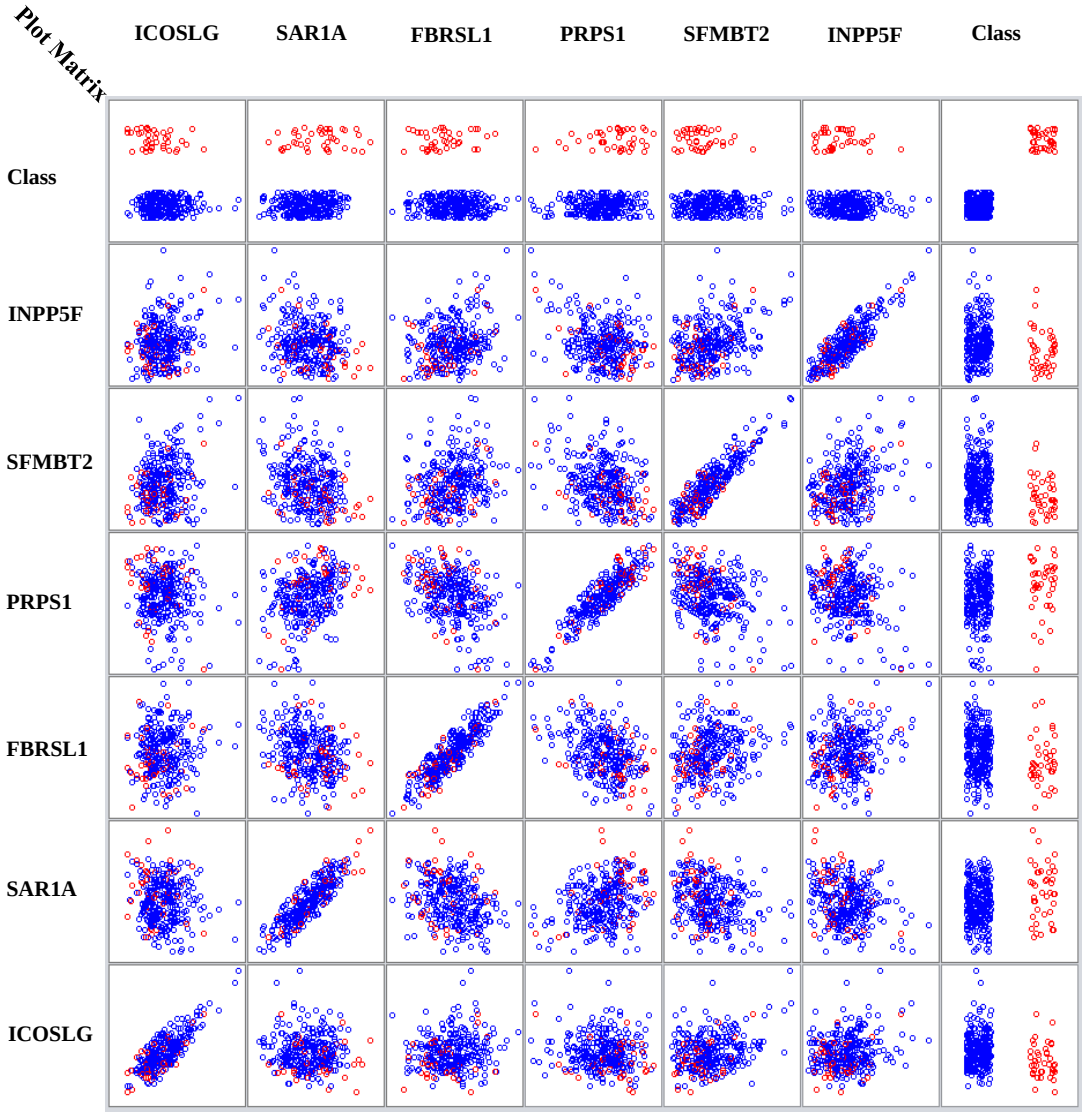

Figure 41. Node Four DS vs. Rest with six genes relations matrix.

In the first node, FGF16 gene is a member of the fibroblast growth factors (FGFs) family, which is involved in a variety of cellular processes, such as stemness, proliferation, anti-apoptosis, drug resistance, and angiogenesis (Katoh et al. 2014). Figure 47 shows that the gene expression of FGF16 is up-regulated and the gene expression of UPF3 is down-regulated in the DH samples compared to the rest of the samples. UPF3 is a regulator of nonsense transcripts homolog B (yeast). Kechavarzi et al. found that UPF3 is one of the actively-upregulated RNA-binding proteins identified in nine cancers in humans and their cancer relevant references, and breast cancer is one of them (Kechavarzi et al. 2014).

In the second node, ASAP1 is shown to be a breast cancer biomarker; it is precisely correlated to its invasive phenotypes that have not been accurately identified (Sabe et al. 2009). Sabe et al. reported that ASAP1 is abnormally overexpressed in some breast cancers and used for their invasion and metastasis. As shown in Figure 42, ASAP1 has a strong coefficient correlation with FBXO41 in the DR samples, but it is less correlated with the remaining samples, as shown in Figure 43. Figure 48 shows that the

genetic expression of ASAP1 is down-regulated in the DR samples compared to the remaining samples. FOSB is a member of the AP-1 family of transcription factors. Bamberger et al. concluded that sharp differences in the expression pattern of AP-1 family members are present in breast tumours, and fosB might be involved in the pathogenesis of these tumours (Bamberger et al. 1999). As shown in Figure 44, FOSB has a strong correlation coefficient with AL71228 in the DR samples, but it was not found to be correlated to the remaining samples, as shown in Figure 45.

In the third node, the VAMP4 gene is a target for some cellular and circulating miRNAs in neoplastic diseases, such as miRNA-31. In any case, it has been confirmed that cellular miRNAs are involved in the development of breast cancer (Allegra et al. 2012). As shown in Figure 44, VAMP4 has a strong coefficient correlation with ARFGAP2 in the LH samples, but it is less correlated to the rest of the samples, as shown in Figure 45. *Figure 46* shows that the genetic expression of VAMP4 is down-regulated in the LH samples compared to the remaining samples while the gene BLP is up-regulated in the LH samples compared to the remaining samples. CT47A1 is one of seven cancer/testis genes in the CT class. CT genes are significantly overexpressed in ductal carcinoma in situ DCIS (Caballero et al. 2014).

In the fourth node, Phosphoribosyl pyrophosphate synthetase 1 (PRPS1) was found to be a direct target of miR124 in breast cancer (Qiu et al. 2015). Nam et al. stated that ICOSLG is a potential biomarker of trastuzumab resistance in breast cancer, which affects the progression of the disease (Nam et al. 2015).

Regarding the fifth node, Dombkowski et al. studied several pathways in breast cancer. They found that ARPC3 reveals extensive combinatorial interactions that have significant implications for its potential role in breast cancer metastasis and therapeutic development (Dombkowski et al. 2011). Zinc finger protein 91 homolog ZFP91 is a methylated target gene in mice. It was identified through methylated-CpG island recovery assay-assisted microarray analysis (Tommasi et al. 2009).

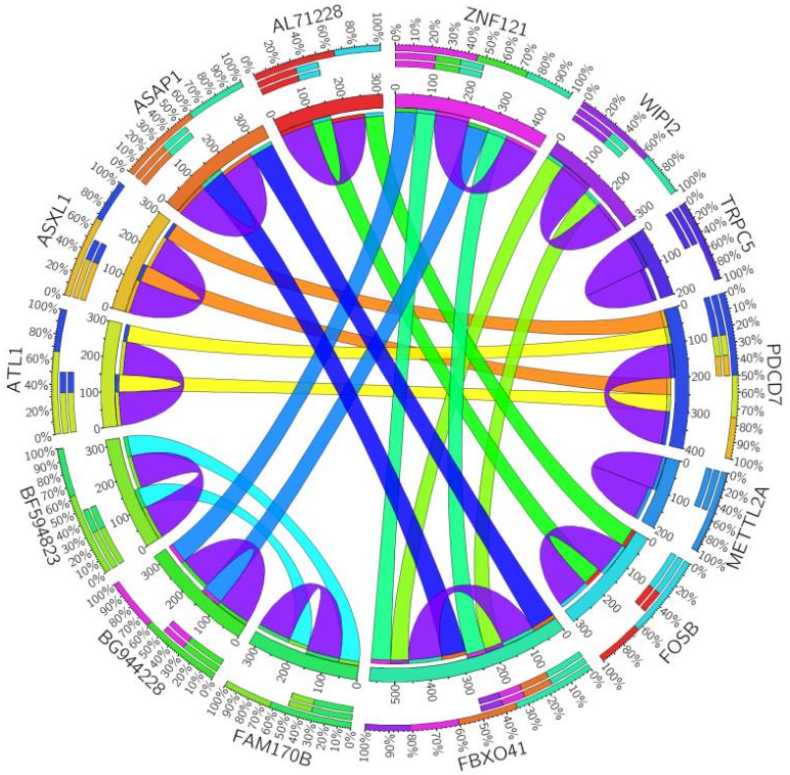

Figure 42. Circos plot for the biomarker genes in node number two for the DR samples based on the correlation coefficient among genes expressions ($p<0.05$).

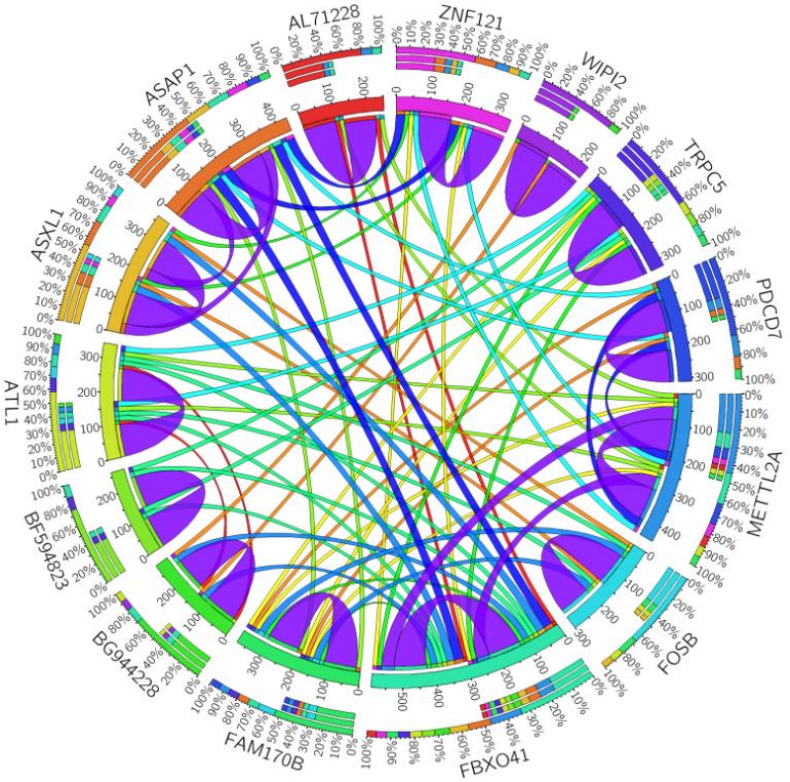

Figure 43. Circos plot for the biomarker genes in node number two for the Rest samples based on the correlation coefficient among genes expressions ($p<0.05$).

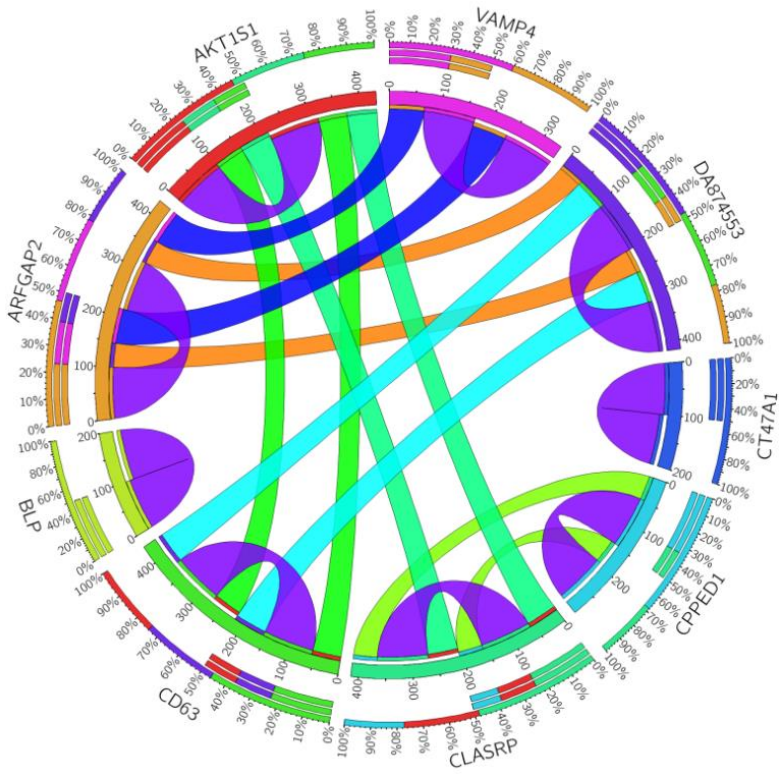

Figure 44. Circos plot for the biomarker genes in node number three for the LH samples based on the correlation coefficient among genes expressions ($p<0.05$).

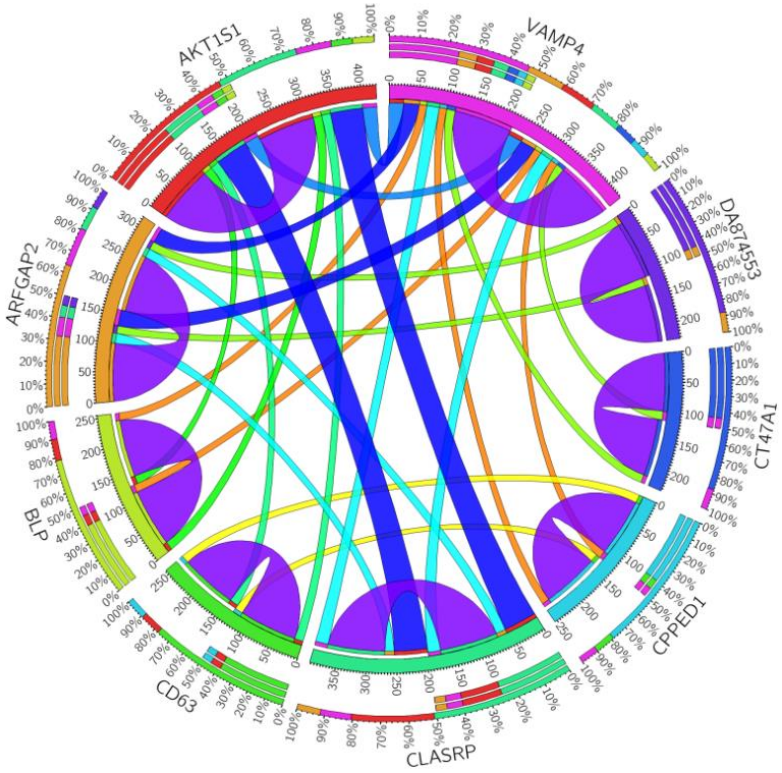

Figure 45. Circos plot for the biomarker genes in node number three for the Rest samples based on the correlation coefficient among genes expressions ($p<0.05$).

Figure 46, Figure 47, and Figure 48 show three of the five nodes for each class against the rest boxplot for the gene biomarkers. The plots also show the up-regulated and down-regulated genes. Most of the biomarkers exhibit clear discrimination between the expression values for a specific class sample versus the remaining samples in the classification node. Many of those biomarkers have outliers, and some of those outliers' values are in the opposite direction of other class, such as the outliers for the UPF3B gene in the "Rest" class versus the "DH" class in the first node, as shown in Figure 47. Some others are in the same direction as those of the other class, such as the outliers for the ZNF121 gene in the "Rest" class versus the "DR" class in the second node, as shown in Figure 48.Some have outliers in both directions, such as the outliers for the ARFGAP2 gene in the "Rest" class versus the "LH" class in the second node, as shown in Figure 46. The outliers that are in the same direction do not interfere in distinguishing the two classes, even though they may misguide the classifier in other scenarios.

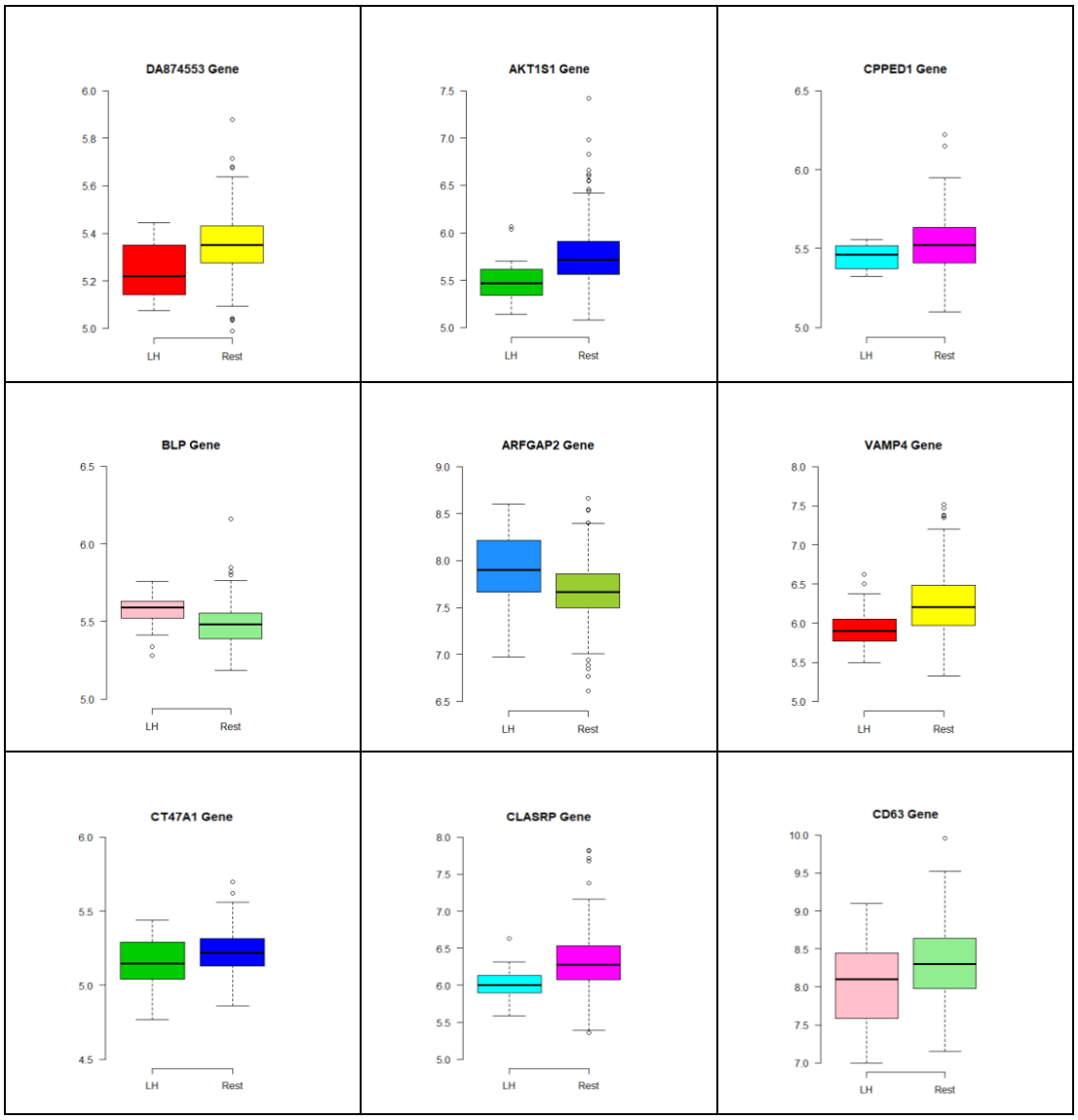

Figure 46. Boxplots for the nine biomarker genes in node number three show the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (LH vs Rest).

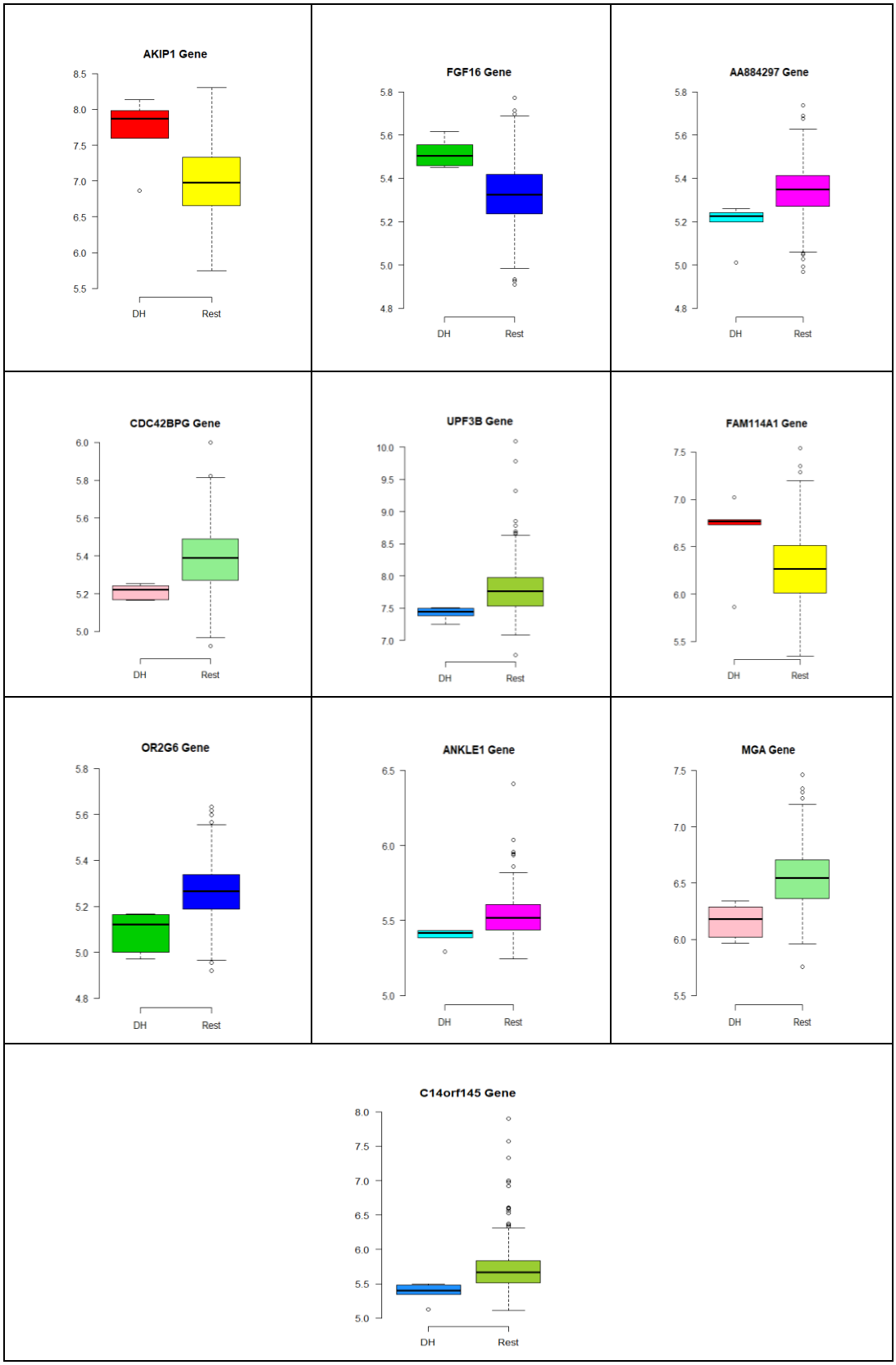

Figure 47. Boxplots for the 10 biomarker genes in node number one show the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (DH vs Rest).

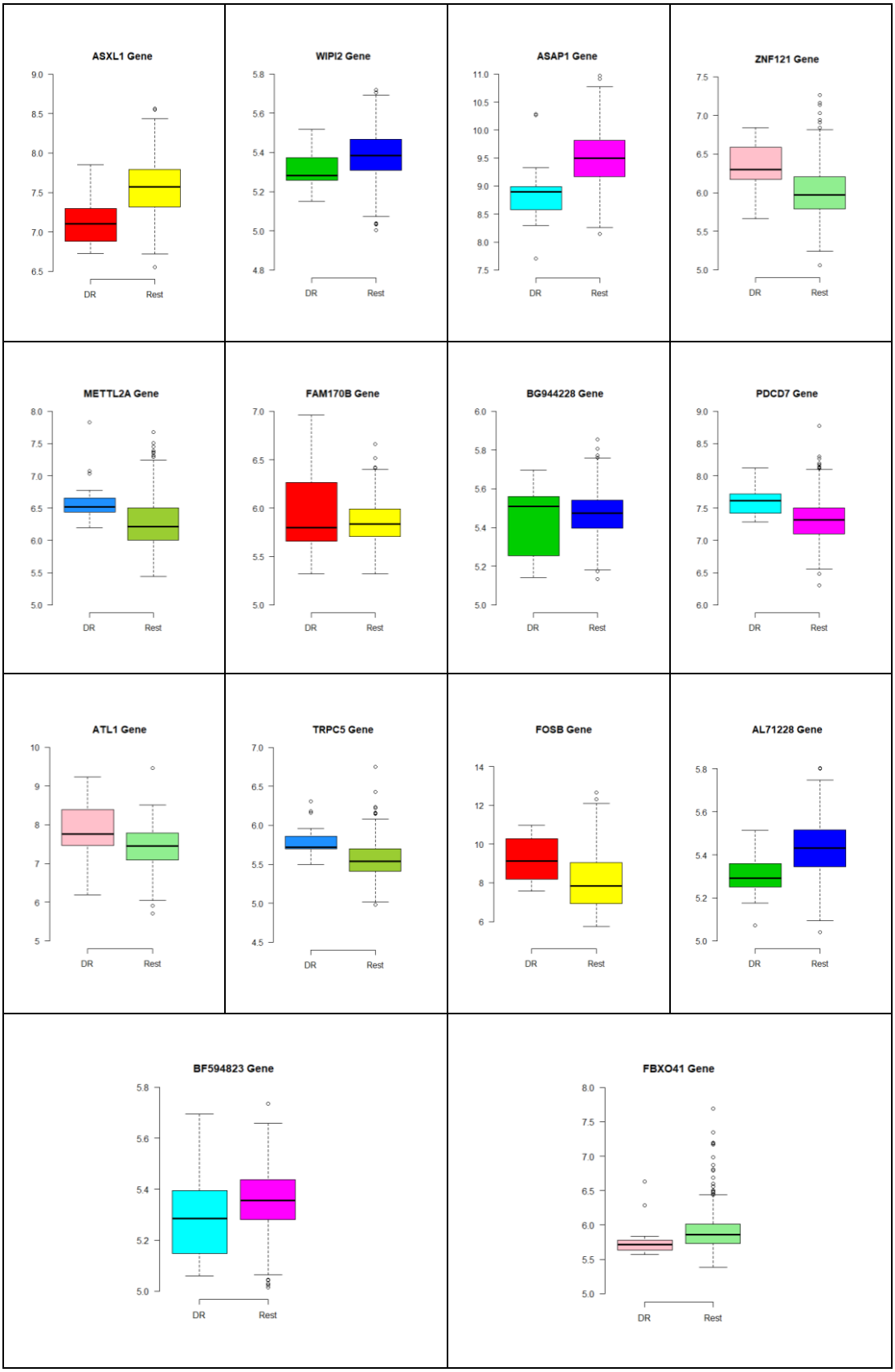

Figure 48. Boxplots for the 14 biomarker genes in node number two show the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (DR vs Rest).

5.2.6 Conclusion

The use of a machine learning model for identifying gene biomarkers for breast cancer survival is a significant step in determining the proper treatment for each patient and will potentially increase survival rates. This study analyzes the gene activities of the survival versus deceased for each therapy, and the potential biomarkers will help to identify the best therapy for the patients based on their gene expression test. This model has very high accuracy levels, and it uses a hierarchical model as a tree that includes one-versus-rest classifications.

The computational model pulls sets of biomarkers for patients who received different treatments. These biomarkers can be used to distinguish whether the patient survived or died in a five-year time window for specific treatment therapy. Related literature was used to verify the relationships between these biomarkers and breast cancer survivability.

Future work includes testing these gene biomarkers in biomedical labs. This novel model can be improved to be used to identify the proper biomarker genes (signature) for different cancer types or even in cases in which patients need or have received more than one type of therapy. Considering additional patient data will enable researchers to cover all missing treatments. With this considerable data size, big data tools, such as Hadoop and Spark, can be utilized to devise an enhanced model.

5.3 Case Study 3

A novel approach for identifying relevant genes for breast cancer survivability on specific therapies. (Tabl et al. 2018)

5.3.1 Abstract

Analyzing the genetic activity of breast cancer survival for a specific type of therapy provides a better understanding of the body responds to the treatment, and helps select the best course of action and while leading to the design of drugs based on gene activity. In this work, we utilize supervised and non-supervised machine learning methods to deal with a multi-class classification problem in which we label the samples based on the combination of the 5-year survivability and treatment; we focus on hormone therapy, radiotherapy, and surgery. The proposed non-supervised hierarchical models are created to find the highest separability between combinations of the classes. The supervised model consists of a combination of feature selection techniques and efficient classifiers used to find a potential set of biomarker genes specific to response to therapy. The results show that different models achieve different performance scores with accuracies ranging from 80.9% to 100%. We have investigated the roles of many biomarkers through the literature and found that some of the discriminative genes in the computational model such as ZC3H11A, VAX2, MAF1, and ZFP91 are related to breast cancer and other types of cancer. MARK2 and ROBO1 both are found in the same classification and in several pathways.

5.3.2 Background

Breast cancer has a very high 5-year relative survival rate (90%) compared to other cancers, including pancreas (8%), lung (18%), and liver (18%). However, breast cancer still accounted for 30% all new cancer cases in women in 2015; furthermore, it is the leading cause of cancer death for women from ages 20 to 59 years in the United States (Miller et al. 2016).

A gene signature in cancer as a predictor for treatment and survival was investigated in earlier works (Chiaretti et al. 2004) (Van De Vijver et al. 2002), in which Chiaretti et al. proposed a non-supervised model in 33 adult patients with T-cell acute lymphocytic leukemia (T-ALL). They found that a single gene interleukin 8 (IL-8), is strongly associated with resistance to first-line treatment and that three genes (CD2, TTK, and AHNAK) are highly predictive of outcome in uniformly treated adults with T-ALL (Chiaretti et al. 2004). De Vijver et al. used a multivariable Cox regression analysis model on a database of 295 breast cancer patients who have a gene-expression signature associated with poor versus

prognosis. They found that the prognosis profile was a key predictor of the clinical outcome (Van De Vijver et al. 2002).

Chang et al. obtained a wound-response signature from 295 early breast cancer patients. They assume that the features of the molecular program of normal wound healing might play a key role in cancer metastasis. The proposed method investigates those signature genes expression in cancer patients. They found that both overall survival and distant metastasis-free survival are markedly diminished in patients whose tumours expressed the wound-response signature compared to tumours that did not express this signature. A gene expression centroid of the wound-response signature acts as a prospectively assigning a prognostic score. Unsupervised hierarchical clustering ("molecular subtypes") and supervised predictors of metastasis ("70-gene prognosis signature") established prognostic signatures. It also suggested that the wound-response signature improves risk stratification independently of known clinicopathologic risk factors (Chang et al. 2005).

Pederson et al. employed a genetics specialist embedded within a multidisciplinary breast clinic, and studied the hereditary cancer risk to assist the decision-making in the cancer treatment. The study focuses on accelerating the surgery based on genetic information. That model was used to compare cancer care between 471 patients in 2012 and 440 patients in 2014; Before embedding a genetic counsellor and the following intervention, the results show that genetic counselling has influenced time to treatment in the 2014 cohort of patients. Recommendation for surgery such as bilateral mastectomy is made for women with mutations in TP53 and PTEN (Pederson et al. 2018).

In this work, we extend an earlier method (Abou Tabl et al. 2017) that was used to predict the proper treatment therapy for better survivability, which is based on gene expression data in breast cancer by handling the multi-class problem using a greedy method of a one-versus-the-rest classification model. In our earlier model, the survival periods of the patients vary, while in the proposed model, the only patients are considered to be survived who lived for more than five years after receiving the treatment. We propose a hierarchical clustering approach based on Ward's linkage to find better borders among the groups of different classes. We then apply standard classifiers on these clusters. The proposed method suggests that for treatment of breast cancer based on gene expression, the model categorizes the survivals and deaths because of breast cancer for each type of treatment by analyzing the genes that can distinguish these classes.

5.3.3 Materials and Methods

Samples from a publicly accessible dataset of 2,433 breast cancer patients and survivors are used in this approach (Pereira et al. 2016), After analyzing the given data, six classes were identified as the baseline of this work. These classes are the combination of each treatment (Surgery, Hormone therapy, Radiotherapy) with a patient status (Living or Deceased) The number of samples (patients) for each class are shown in Table 9, which indicates that a total of 347 patients are used in this work.

Table 9. Class list with the number of samples in each class.

Class	Number of samples
Living & Radio (LR)	132
Deceased & Radio (DR)	19
Living & Hormone (LH)	20
Deceased & Hormone (DH)	6
Living & Surgery (LS)	130
Deceased & Surgery (DS)	40
Total	**347**

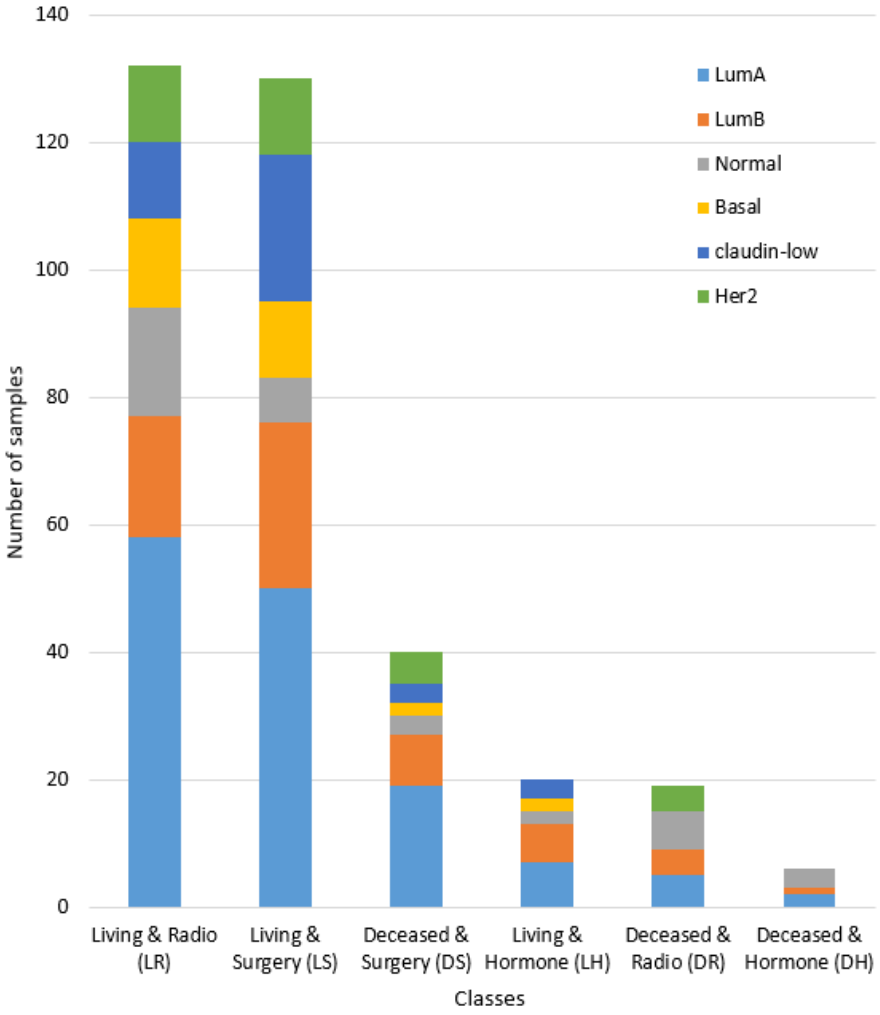

Figure 49. The distribution of breast cancer subtypes samples in each class.

Figure 49 depicts the distributions of the breast cancer subtypes samples in each class. The subtypes are well-distributed in each class, at least three subtypes are represented in each class, which means that the possibility of a correlation between subtypes and classes is very low.

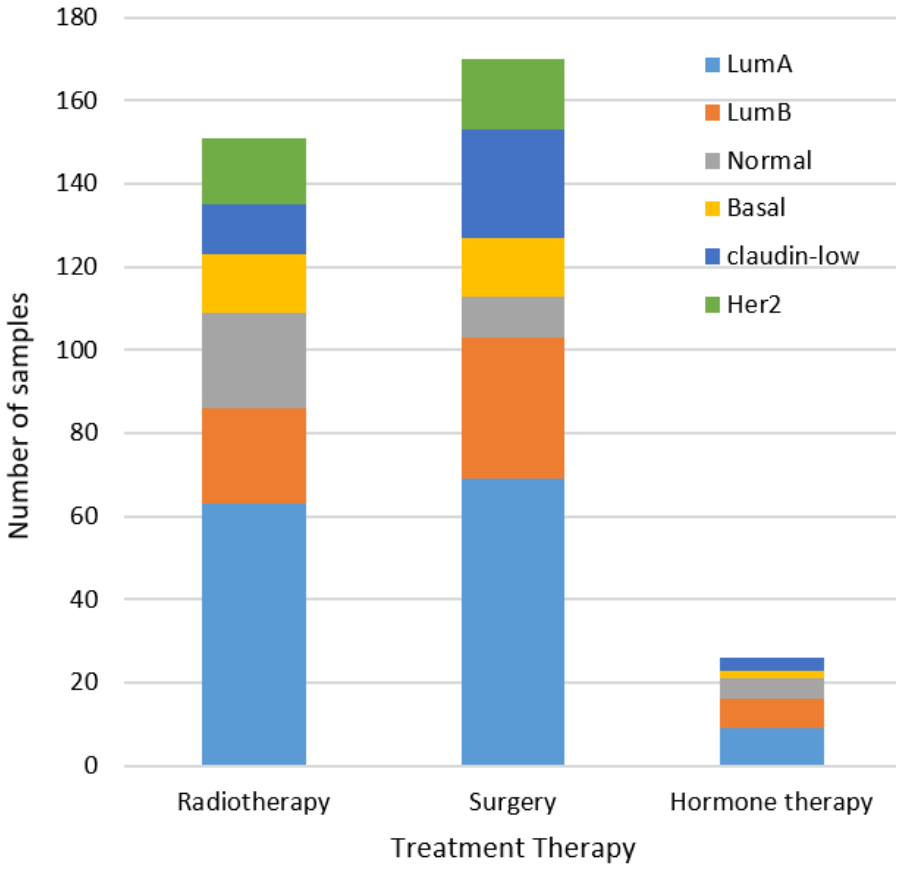

Figure 50. The distribution of breast cancer subtypes samples in each treatment therapies samples.

Based on the available data, only three treatment therapies are covered; they are Surgery, Hormone therapy, Radiotherapy (Figure 50). Our proposed model is a bottom-up hierarchical multiclass tree obtained using agglomerative clustering technique. The dataset contains imbalanced classes, a problem that is well-known in machine learning. The pipeline of the proposed model starts with feature selection methods, including Chi-square (Mantel 1963) and Info-Gain, which are applied for limiting the significant number of features (genes). A wrapper method is also used to obtain the best subset of genes that represent the model by utilizing mRMR (minimum redundancy maximum relevance) feature selection method (Peng et al. 2005). This was followed by applying several class balancing techniques such as SMOTE (Chawla et al. 2002), cost-sensitive (Núñez 1988), and resampling (Gross 1980) to balance the number of classes before applying different types of classifiers such as Nave Bayes (Domingos et al. 1997) and random forest (Breiman 2001). Finally, a small number of biomarker genes are identified for predicting proper treatment therapy. To the best of our awareness, this work is the first prediction model, which is built on the combination of treatment and survivability of the patient as a class.

The patient class distribution is shown in Figure 51, which depicts the percentage of samples within each class. It is clear that there are significant differences between the number of samples of the different classes, which requires class balancing to achieve a fair calcification.

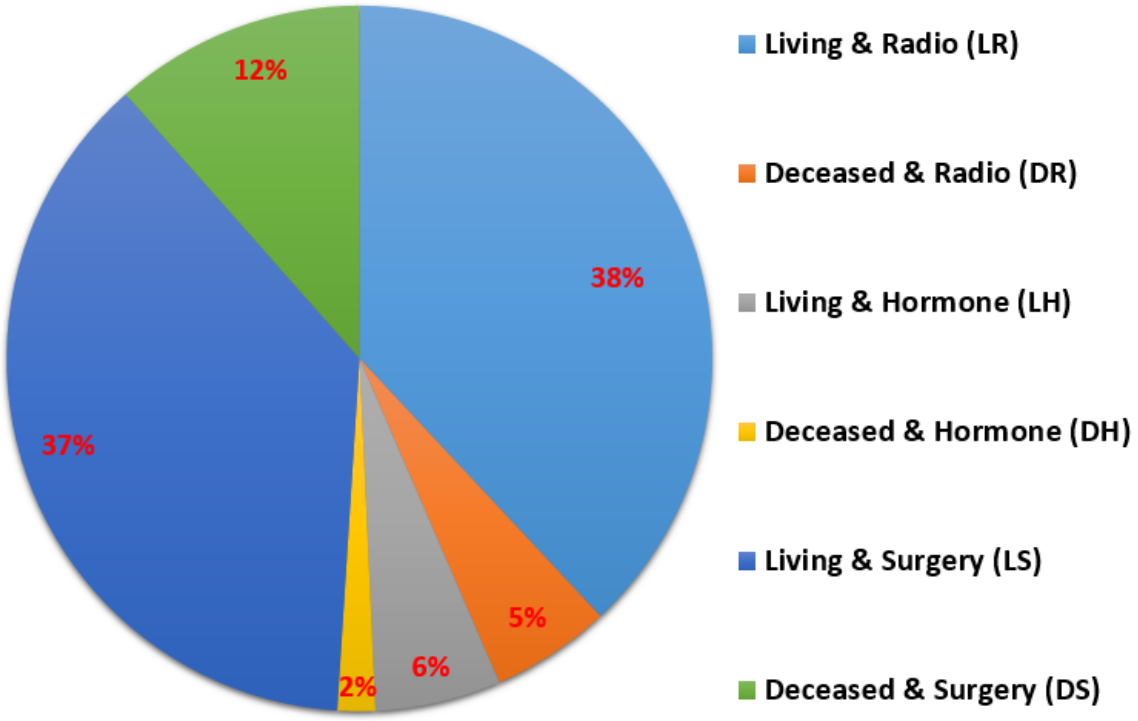

Figure 51. Percentages of patient class distribution.

5.3.4 The bottom-up Multi-class Classification approach

In our proposed bottom-up approach, we build five models based on the linkage type between classes. We start with six distinct datasets of samples responding to the six classes and then build a bottom-up fashion tree. The flowchart is illustrated in Figure 52, which shows the steps for obtaining the five models based on the distance between the classes.

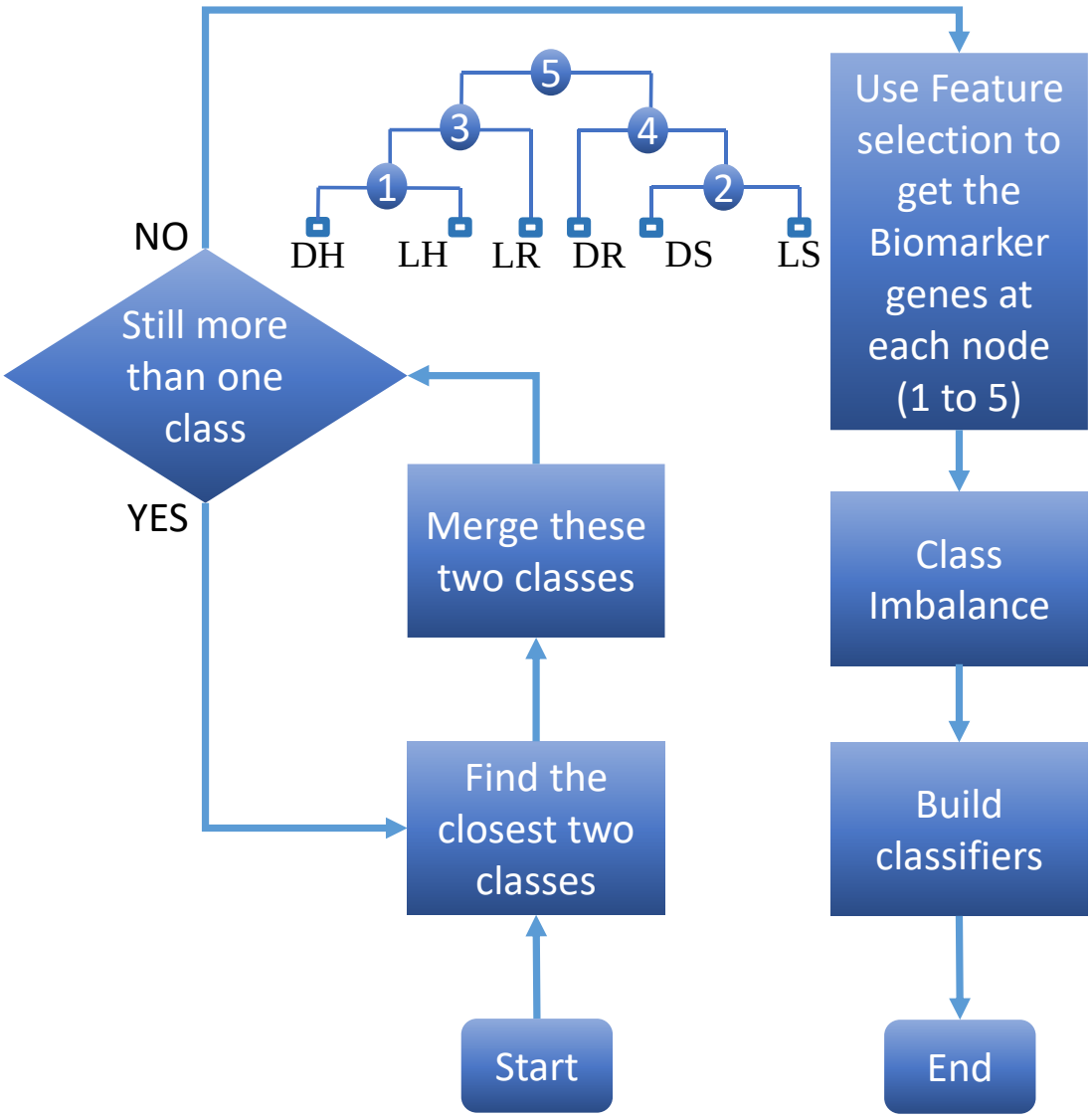

Figure 52. Schematic representation of the proposed models based on the linkage type.

In the first step, the distance matrix between all pairs of the six classes is calculated. Then, the two classes i and j with the minimum distance di,j are merged. As a result, we obtain a new distance matrix after merging the two closest classes (five classes), and the two classes with the minimum distance are merged until we obtain only a single class.

The merging steps in the model are shown in Table 10. Step one shows the distance matrix between the six classes. In step two, classes C1 and C4 are merged since d1,4 is the smallest distance in the table, the two classes are merged and form a new dataset, which is the combination of the samples from these two classes. For simplicity, we call it class C14. In step 3, these new three classes are compared again in a pairwise fashion until only one class remains in step five.

Table 10. Computing the distance between each pair of classes; di,j is the distance between classes i and j.

Step1: distance matrix between six classes

	C_2	C_3	C_4	C_5	C_6
C_1	$d_{1,2}$	$d_{1,3}$	$d_{1,4}$	$d_{1,5}$	$d_{1,6}$
C_2		$d_{2,3}$	$d_{2,4}$	$d_{2,5}$	$d_{2,6}$
C_3			$d_{3,4}$	$d_{3,5}$	$d_{3,6}$
C_4				$d_{4,5}$	$d_{4,6}$
C_5					$d_{5,6}$

Step 2: merging C_1 and C_4

	C_3	C_5	C_6	C_{14}
C_2	$d_{2,3}$	$d_{2,5}$	$d_{2,6}$	$d_{2,14}$
C_3		$d_{3,5}$	$d_{3,6}$	$d_{3,14}$
C_5			$d_{5,6}$	$d_{5,14}$
C_6				$d_{6,14}$

Step 3: merging C_{14} and C_3

	C_5	C_6	C_{143}
C_2	$d_{2,5}$	$d_{2,6}$	$d_{2,143}$
C_5		$d_{5,6}$	$d_{5,143}$
C_6			$d_{6,143}$

Step 4: merging C_{143} and C_6

	C_5	C_{1436}
C_2	$d_{2,5}$	$d_{2,1436}$
C_5		$d_{5,1436}$

Step 5: merging C_{1436} and C_5

	C_{14365}
C_2	$d_{2,14365}$

The distance matrix used in this work is the Euclidean distance. The Euclidean distance between two classes $X = \{x_1, x_2, x_3, \dots \dots x_n\}$ and $Y = \{y_1, y_2, y_3, \dots \dots y_n\}$ is defined as follows:

$$d = |x - y| = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2} \qquad (10)$$

There are several linkage methods to compute the distance between two clusters. Figure 53 shows some approaches that can be used, such as single linkage, complete linkage, average linkage, centroid linkage, and Ward's linkage methods. Both single and complete linkages types rely on a pair of samples for determining the distance between two clusters, while the other three linkage types, average linkage, centroid linkage, and Ward's linkage rely on all samples within each class for determining the distance between the classes.

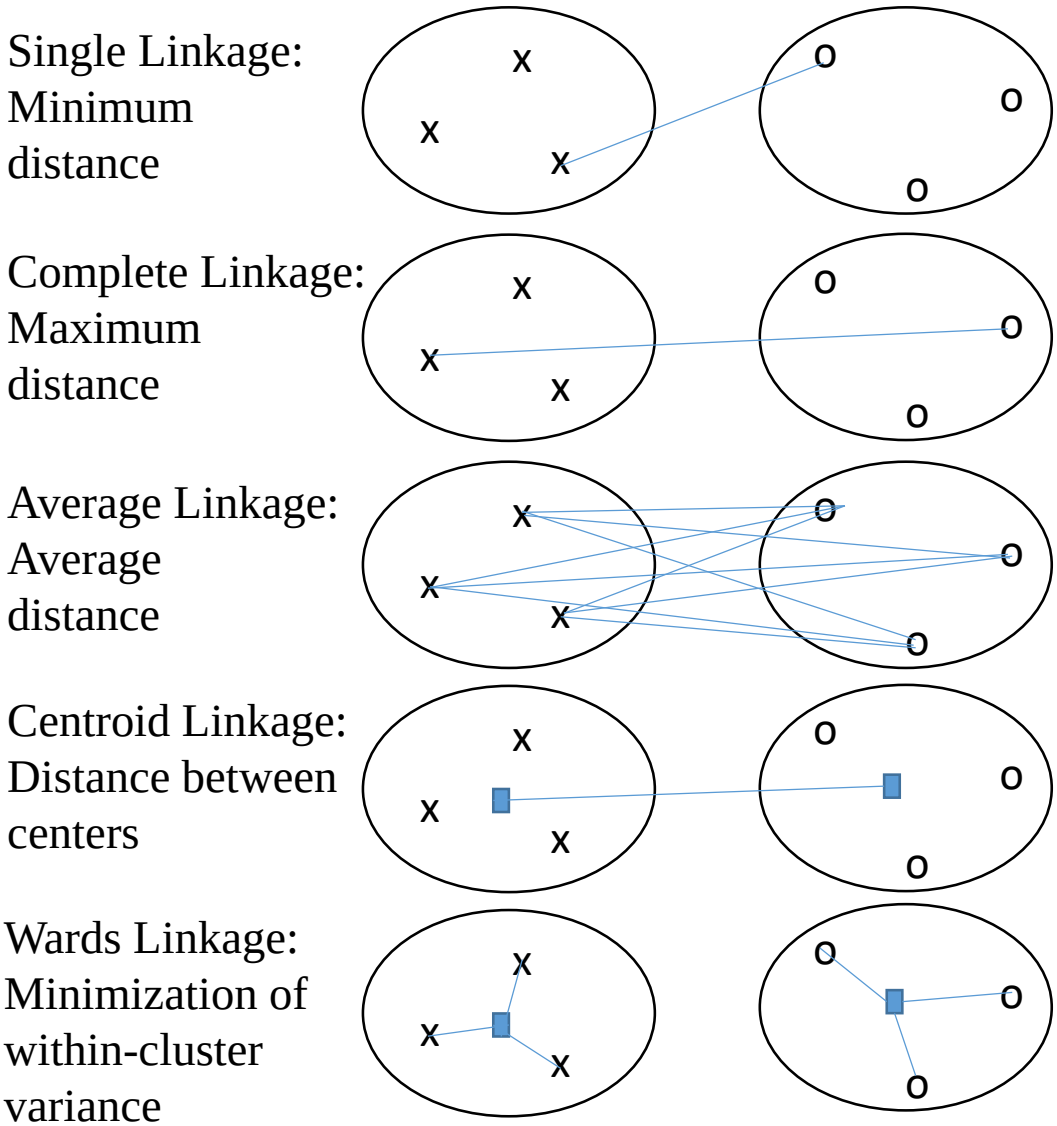

Figure 53. The five linkage types: Single, Complete, Average, Centroid, and Ward's linkage.

Single linkage the distance between two clusters is the distance between the two nearest neighbour's samples in such a way that two neighbours belong to different clusters. This can be formulated as follows:

$$d(C_i, C_j) = \min_{x \in C_i, y \in C_y} d(x, y) \quad (11)$$

Complete linkage evaluates the distance between two clusters based on the distance between the furthest neighbours, where each neighbour belongs to one of the clusters. This can be formulated as follows:

$$d(C_i, C_j) = \max_{x \in C_i, y \in C_y} d(x, y) \quad (12)$$

Average linkage, on the other hand, takes the average of the distances between all pairs of samples into account. In other words, the distance between two clusters using the average linkage method can be computed as follows:

$$d(C_i, C_j) = \frac{1}{|C_i| \times |C_j|} \sum_{x \in C_i} \sum_{y \in C_y} d(x, y) \quad (13)$$

Centroid linkage uses the distance between the centroids of the two classes

$$d(C_i, C_j) = d(\overline{x_i}, \overline{y_J}) \qquad \textit{Where} \qquad \overline{x_i} = \frac{1}{n_i} \sum_{i=1}^{n} x_i \quad (14)$$

Ward's linkage is one of the other approaches that use analysis of variance to evaluate the distances between clusters (Johnson 1967). Ward's linkage minimum variance method is a special case of the objective function approach initially presented in (Ward Jr 1963). Ward's linkage works as follows:

Using analysis of variance (Anova) to evaluate the distances between clusters.

Minimizing the sum of squares of any two (hypothetical) clusters that can be formed at each step, as follows:

$$d_{ij} = \frac{N_i \times N_j}{N_i + N_j} \sqrt{||c_i - c_j||^2} \quad (15)$$

Where Ni and Nj are the numbers of samples in cluster i and j respectively, and Ci and Cj denote the centers of the clusters; ||.|| is the Euclidean norm.

• The mean and cardinality of the newly merged cluster, k, is computed as follows:

$$c_k = \frac{1}{N_i + N_j} N_i c_i + N_j c_j \quad (16)$$

$$N_k = N_i + N_j \quad (17)$$

5.3.5 Feature Selection

The gene expression dataset contains 24,368 genes for each of the 347 samples. The problem of the curse of dimensionality makes it difficult to classify the dataset in its current form. Hence, feature selection is essential to narrow down the number of genes to a few genes at each node. Chi-square and Info-Gain are applied to select the best information gain of the selected genes, then mRMR (minimum redundancy maximum relevance) feature selection is applied to find the best subset of significant genes. mRMR is an algorithm commonly used in a greedy search to identify characteristics of features and narrow down their relevance correctly.

5.3.6 Class Imbalance

These five models utilize one-versus-rest to handle the multiclass problem, which leads to an unbalanced class dataset at each node of the classification model. Therefore, we applied several techniques to handle this issue, such as:

- **Oversampling**: Oversampling the minority class by using synthetic data generators. There are several algorithms to achieve this; we used one of the most popular algorithms, Synthetic Minority Over-Sampling Technique (SMOTE).
- **Cost-sensitive classifier**: Using penalized models that apply additional costs for the minority class to achieve class balancing. This, in turn, bias the model to pay more attention to the minority class. The algorithm used in this work is called Cost-Sensitive Classifier in Weka using a penalty matrix to overcome the imbalance.
- **Resampling**: Replicating the dataset, which can be done by one of two methods. First, adding more copies of the data instances to the minority class, called over-sampling. Second, by deleting some instances of the majority class, called under-sampling. We used the oversampling technique.

5.3.7 Classification

After deriving the five models using the five linkage types to find the closest classes a hierarchical tree obtained using agglomerative clustering. The standard classifiers were applied to determine which biomarker genes are the most discriminative ones in terms of separating the classes in each branch of the tree.

In order to train SVM classifying, libSVM library (Chang 2011) with linear kernel was utilized within a grid search algorithm to optimize the parameters of the classifier. After running the algorithm on the data, we found that Ward's linkage method is the one that achieves better accuracy and most meaningful hierarchy, based on the six classes.

5.3.8 Results and Discussion

The Ward's linkage model shows the best performance measurements than the rest of the models. Moreover, it has a balanced tree of the treatment survival clusters, as shown in Figure 54, which leads to easier maintaining a different group of clusters. Table 11 shows the discriminative genes between each group of clusters in the tree. The results suggested that the separation between the clusters in the lower part of the tree is significantly high-performance scores between 99% - 100% for classifying the tree nodes. The accuracies of classifying nodes are 100% for DH vs. LH, and 99.2% for DS vs. LS. The scores remain high in the middle part of the tree with accuracy 99.6% for the left side which is (DH, LH) vs. LR, 99.5% for the right side which is DR vs. (DS, LS). While the scores drop down on the root of the tree where we classify the left side vs. the right side of the tree to 81.8% for classifying 2 clusters with many classes in each of them. The results for the four models are presented in the supplementary materials.

In Ward linkage, the objective function is based on sum square error, is to minimize the within-cluster variance to improve the classification performance rather than reducing the distance between each pair of clusters.

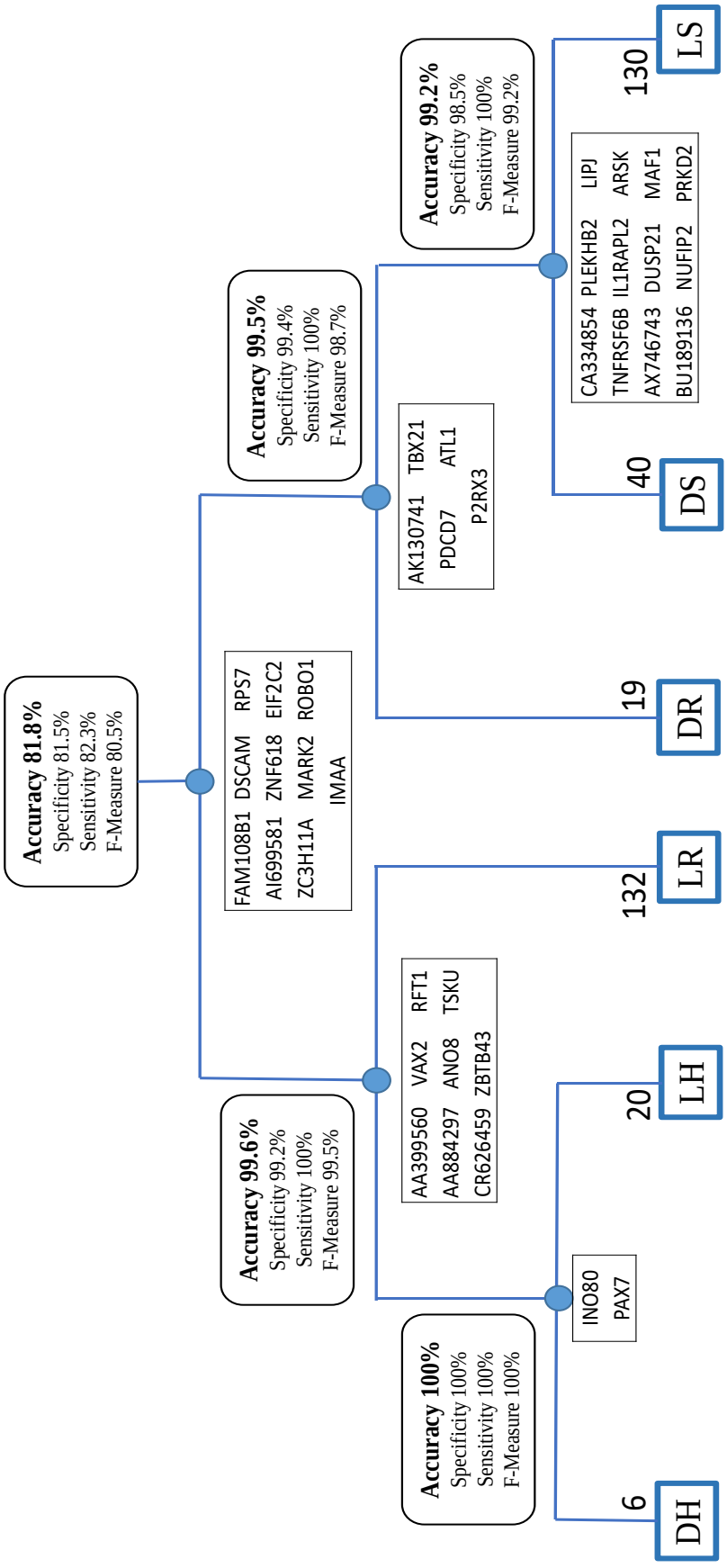

Figure 54. Ward's linkage model: classification model with performance measures.

Table 11. Ward ‘s linkage model: 37 biomarker genes.

	DH vs. LH	DS vs. LS	LR vs. DH_LH	DR vs. DS_LS	LR_DH_LH vs. DR_DS_LS
Genes	INO80	CA334854	AA399560	AK130741	FAM108B1
	PAX7	TNFRSF6B	AA884297	PDCD7	AI699581
		AX746743	CR626459	TBX21	ZC3H11A
		BU189136	VAX2	ATL1	DSCAM
		PLEKHB2	ANO8	P2RX3	ZNF618
		IL1RAPL2	ZBTB43		MARK2
		DUSP21	RFT1		RPS7
		NUFIP2	TSKU		EIF2C2
		LIPJ			ROBO1
		ARSK			IMAA
		MAF1			
		PRKD2			

Figure 55 shows a multi-dimensional representation of the plot matrix for the five discriminative genes found in Ward’s linkage model for the node of DR class versus (DS, LS) class, as an example; the figure also shows the relations among the five genes with each other. It is clear that from the class column, the samples are separable with not much overlapping for the 2 clusters.

Figure *56* shows the boxplot for some biomarker genes which indicates the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (DH vs. LH) and (DR vs. (DS, LS)). The gene expression of INO80 is slightly up-regulated in the DH samples comparing to the LH of the samples, TBX21 is also up-regulated in the DR samples comparing to the DSLS of the samples. While it shows that the gene expression of PAX7 is down-regulated in the DH samples comparing to the LH of the samples, AK130741 is also down-regulated in the DR samples comparing to the DSLS of the samples.

For Ward’s Linkage model for the “DS” vs. “LS” node and as shown in Figure 57, CA334854 gene has a strong correlation coefficient with two genes AX746743 and IL1RAPL2 in the DS samples, while there is no significant correlation between them in the LS samples as shown in Figure 58.

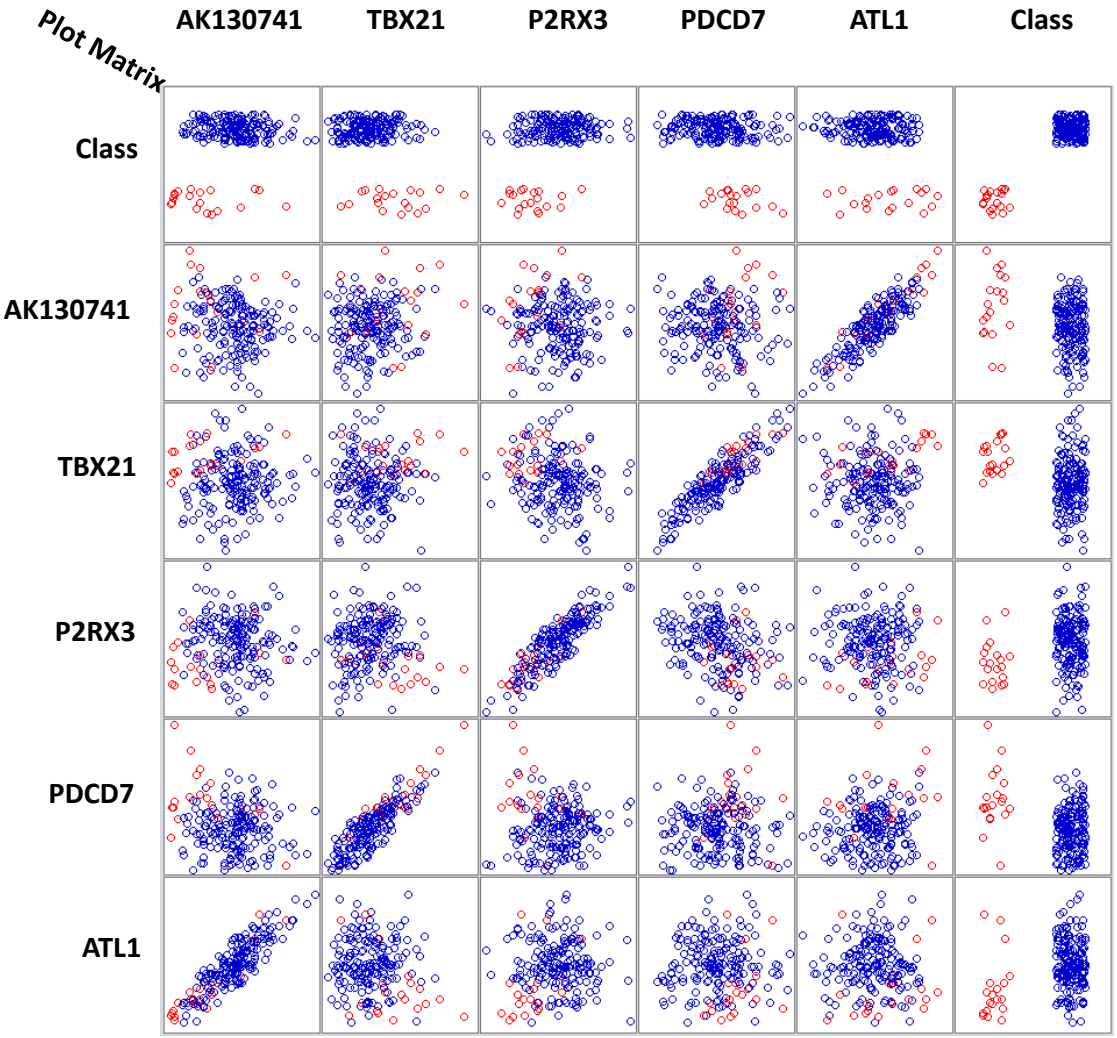

Figure 55. Ward's Linkage model DR vs. (DS, LS) Node with five genes relations matrix.

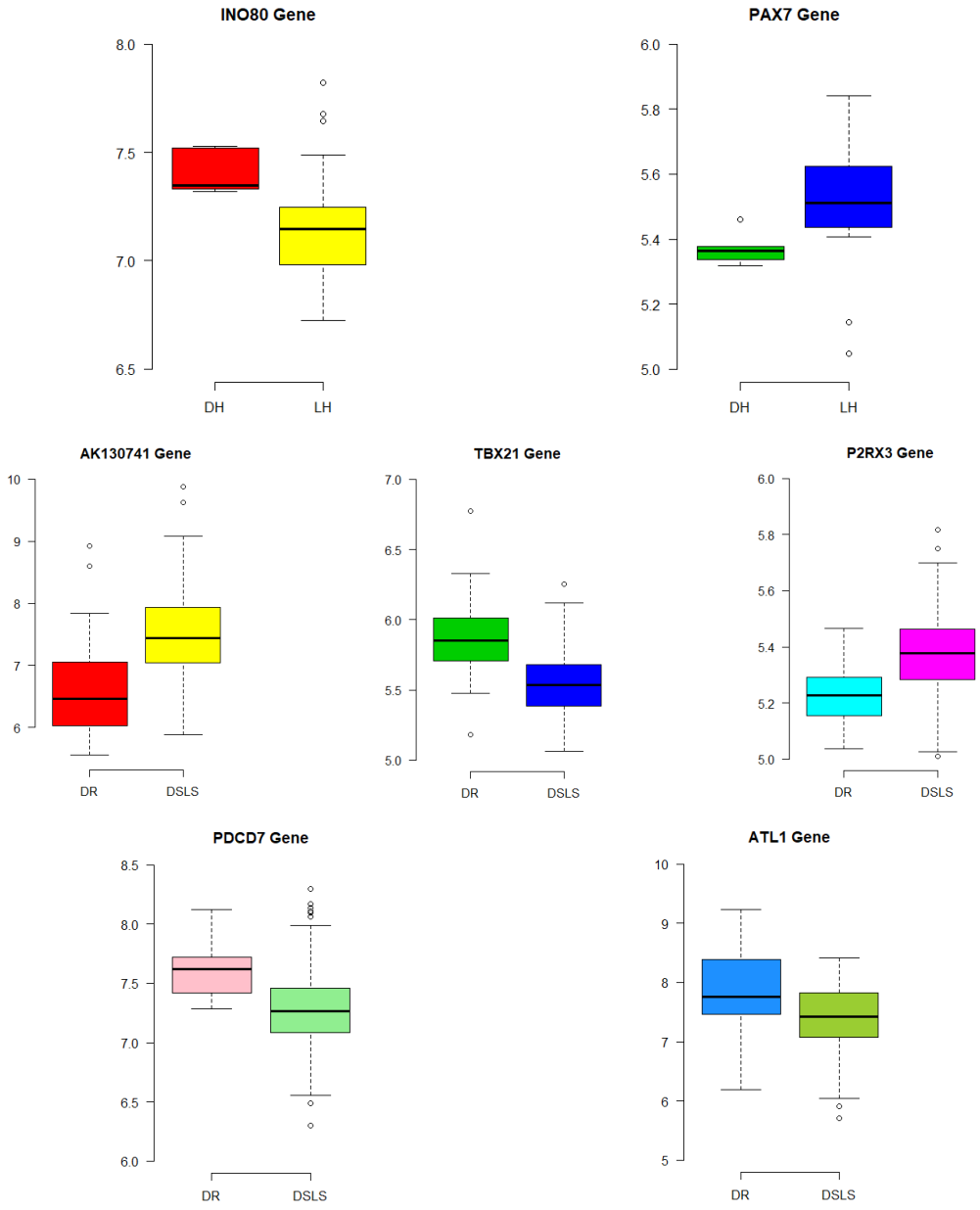

Figure 56. Boxplot for the biomarker genes in Ward's linkage model shows the minimum, first quartile, median, third quartile, and maximum gene expression values for each group of samples (DH vs. LH) and (DR vs. (DS, LS)).

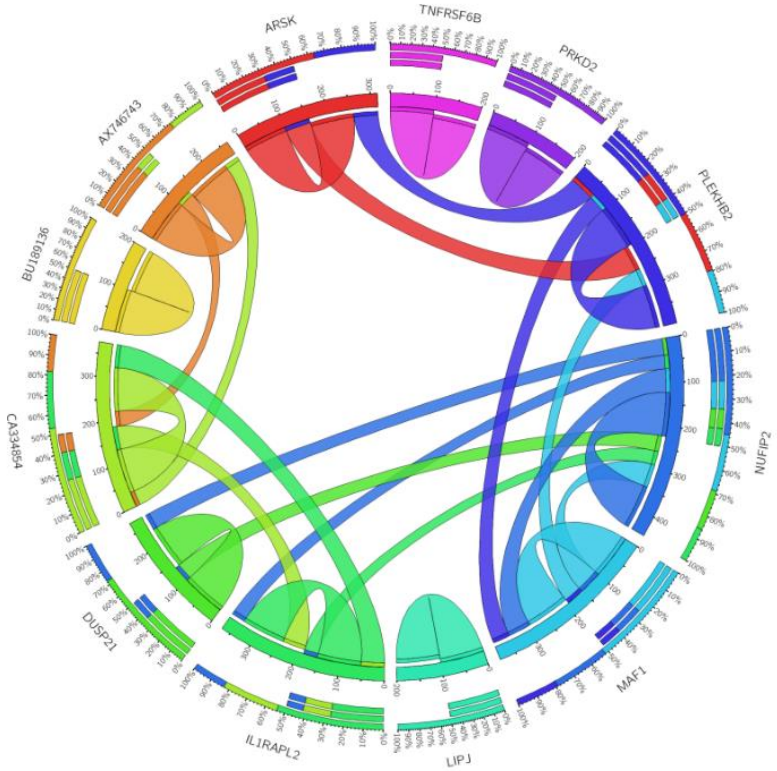

Figure 57. Circos plot for the biomarker genes in Ward's linkage model for the DS class samples based on the correlation coefficient among genes expressions ($p<0.05$).

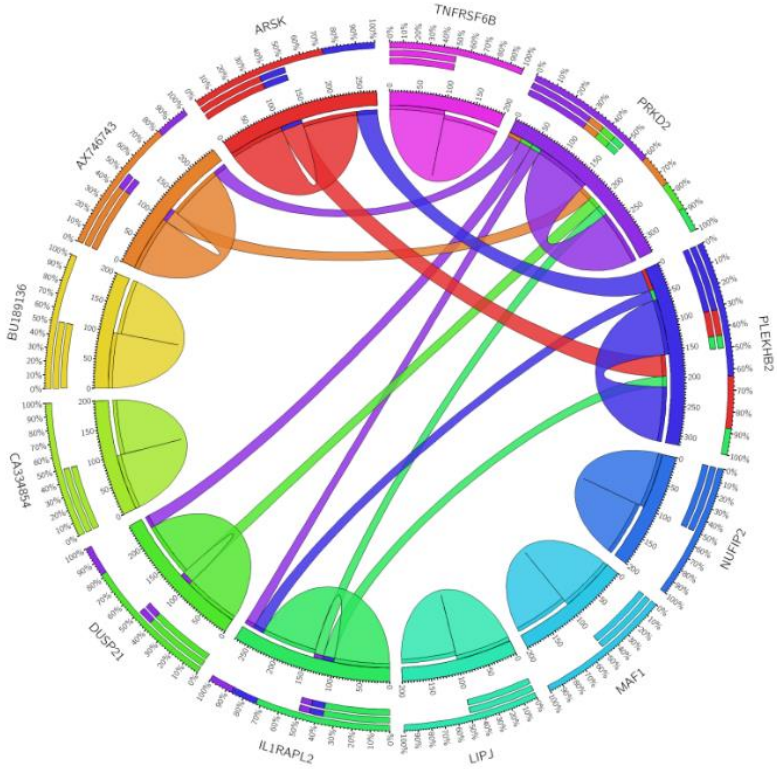

Figure 58. Circos plot for the biomarker genes in Ward's linkage model for the LS class samples based on the correlation coefficient among genes expressions ($p<0.05$).

5.3.9 Biological Insight

For the discriminative genes in DH vs LH node, INO80 and PAX7 genes are both involved in the regulation of epigenetic histone marks and chromatin remodelling (Gošev et al. 2017). As part of the analysis of epigenetic modifications around the INO80 interaction site, Mendiratta et al. studied the NO80 binding region of HOXC11 and PAX7 genes by ChIP with anti-H3K9ac and anti-H3k27me3 followed by qPCR. In both, the cases studied, INO80 enrichment was correlated with H3K27me3 (Mendiratta et al. 2016). Both of them also were reported in a protein-protein interaction network for cancer (Frenkel-Morgenstern et al. 2017).

Some of the fond gene in the computational model are related to breast cancer. Cai et al. studied the identify breast cancer susceptibility loci rs4951011 at 1q32.1 in intron 2 of the ZC3H11A gene; the three-genome study was conducted on patients from the Eastern Asians population, mainly Chinese and Koreans. They also found that Expression levels of the ZC3H11A gene were significantly higher in the tumour tissue than in adjacent normal tissue ($p = 0.0049$) in TCGA data. The function of ZC3H11A is not clear (Cai et al. 2014).

VAX2 is a protein-coding gene that encodes a homeodomain-containing protein from a class of homeobox transcription factors which are conserved in vertebrates (Hallonet et al. 1998). Gu et al. identified the top 40 most correlated genes with similar methylation patterns calculated by Pearson correlation, VAX2 is one of them (Gu et al. 2013). VAX2 is found to be a transcription factor that regulates three genes (PLCB4, ADCY6, CNR1) in RNA tissue in response to chemotherapy in patients with operable breast cancer (Li et al. 2017).

MAF1 displays tumour-suppressor activity. Surprisingly, blocking the synthesis of rRNA and tRNAs is insufficient to account for MAF1's tumour-suppressor function. MAF1 binds to the PTEN promoter to enhance PTEN promoter acetylation and activity. MAF1 down-regulation unexpectedly leads to activation of AKT-mTOR signalling, which is mediated by decreased PTEN expression (Li et al. 2016).

ZFP91 serves as a positive regulator for MAP3K14 gene, causing its stabilization and activation. Overexpression of MAP3K14 has been associated with neoplastic growth such as in melanoma, pancreatic carcinoma, lung cancer, breast cancer, multiple myeloma and adult T-cell leukemia. ZFP91-mediated stabilization may tolerate one of the mechanisms of MAP3K14 oncogenic activation (Paschke et al. 2014).

Labhart et al. identified DSCAM as one of the target genes in breast cancer cells, which are directly regulated by the SRC-3/AIB1 coactivator (Labhart et al. 2005). Stuhlmiller et al. defined a signature of kinases that regulate MARK2, the kinases involved in significant changes for MIB binding after 48-hr lapatinib treatment for breast cancer cells (Stuhlmiller et al. 2015). ROBO1 is a cell adhesion receptor that is a survival and growth factor for breast cancer (Minn et al. 2005). Using cBioPortal (Cerami et

al. 2012), we investigated the pathway of genes on another breast cancer dataset (Network 2012), The three genes (DSCAM, MARK2m, and ROBO1) from node were found connected in the pathway shown in Figure 59. DSCAM, MARK2 was also reported to be in two pathways combined with RPS7 in Reactome pathway knowledgebase (Croft et al. 2013), the two pathways are Axon guidance (R-HSA-422475) and Developmental Biology (R-HSA-1266738). The full information about these pathways and some other pathways in which the biomarkers are involved in them are included in Supplementary pathways.

Two genes from DS vs LS node were also reported in Rectome database, ARSK and PRKD2 were found in three pathways which are Sphingolipid metabolism (R-HSA-428157), Metabolism of lipids (R-HSA-556833), and Metabolism (R-HSA-1430728). See Supplementary pathways for more information.

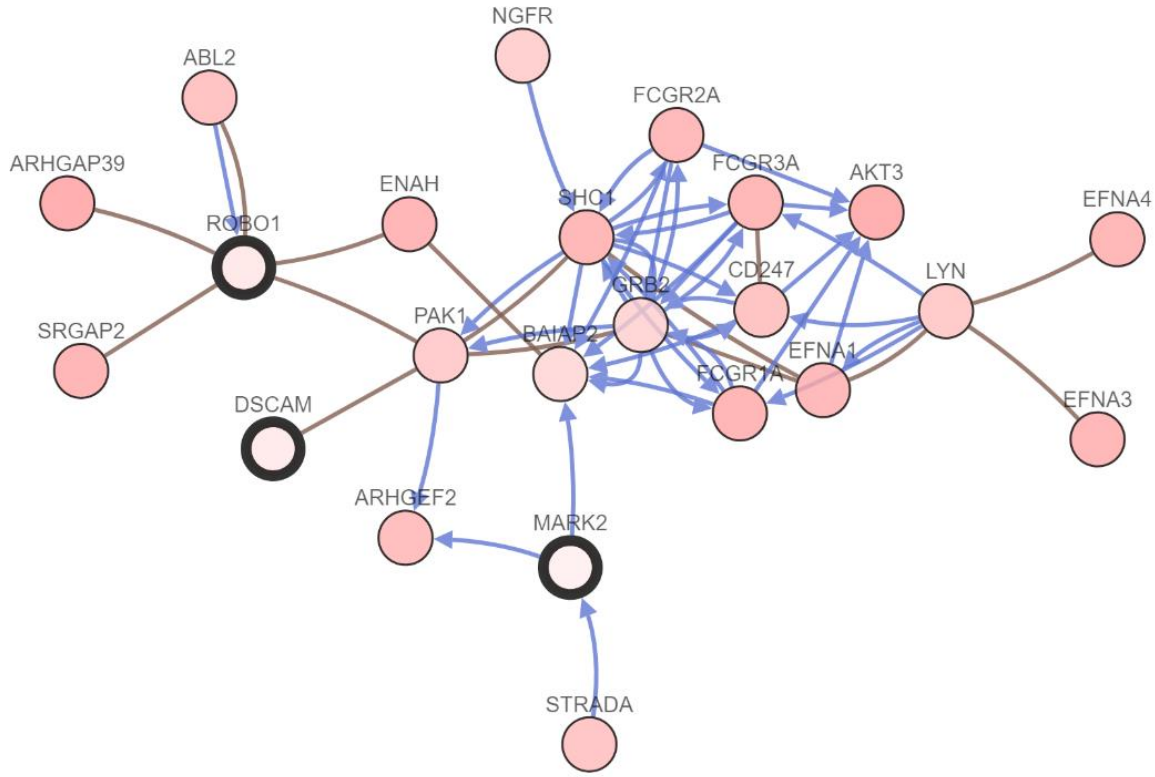

Figure 59. Network genes pathway that includes most frequently altered neighbour genes for (DSCAM, MARK2, ROBO1).

5.3.10 Conclusion

In conclusion, a hierarchical clustering model based on Ward's linkage found to be discriminative in drawing borders for survival treatments classes in breast cancer. Based on the gene expressions data, standard classifiers perform very well in the nodes of the clusters in the constructed hierarchical tree. The results suggest subsets of genes, in which, some of the genes in the same nodes are reported to be related in functions or pathways, and some of them are strongly related to breast cancer. ZC3H11 is highly statistically significant expresses in tumour tissue, VAX2 is associated with the response of chemotherapy in breast cancer, while MAF1 is a tumour suppressor, and ZFP91 is a positive regulator for MAP3K14 that is related with breast cancer. MARK2 and ROBO1 have coexisted in some pathways. Also, ARSK and PRKD2 have the same case.

5.3.11 Supplementary Materials

Table 12. Single linkage model: 41 biomarker genes.

	DR vs. LS	LR vs. DR_LS	LH vs. DR_LS_LR	DS vs. DR_LS_LR_LH	DH vs. DR_LS_LR_LH_DS
Genes	AK130741	ACVR2B	FAM126B	FAM75D5	FAM114A1
	TBX21	DSCAM	BX119171	AI636666	CDC42BPG
	P2RX3	SDHD	COBRA1	CSH2	C14orf145
		VPS26A	VAX2	RIPK1	ANKLE1
		METTL6	TRIB1	SAR1A	OR2G6
		IGF1	N35251	PDXK	UPF3B
		RPS7	ATP10A	BX093437	MGA
		MLL3	Z38762		FGF16
		UPF2	C15orf5		AKIP1
		RUNDC2B	LAMTOR1		AA884297
		CAPN1			

Table 13. Complete linkage model: 31 biomarker genes.

	DH vs. LH	LR vs. DH_LH	DS vs. DH_LH_LR	LS vs. DH_LH_LR_DS	DR vs. DH_LH_LR_DS_LS
Genes	INO80	AA399560	CA429430	ARPC3	RRAGC
	PAX7	AA884297	DA570923	TMEM128	FAM193B
		CR626459	C1orf147	ACVR2B	BG944228
		VAX2	ZNF148	GYPC	LHFPL1
		ANO8	OR2B3	CAPN1	IL28RA
		ZBTB43	ZNF43	X7.Sep	FOS
		RFT1	SYNE1	BX116795	MGC24103
		TSKU			

Table 14. Average linkage model: 34 biomarker genes.

	DH vs. LH	DS vs. DH_LH	LS vs. DH_LH_DS	LR vs. DH_LH_DS_LS	DR vs. DH_LH_DS_LS_LR
Genes	INO80 PAX7	PQLC1 TRUB2 CDC42EP4 SOX18 N53502 TLK1	BF739944 CEP164 PSTPIP2 DKFZp434K1323 ARSK WDR81 PRMT1	TRAPPC10 ZC3H11A DSCAM ZNF579 ARPC3 ACTN1 CNOT3 UBLCP1 PTMA ZFP91 TAC4 PPIB	RRAGC FAM193B BG944228 LHFPL1 IL28RA FOS MGC24103

Table 15. Centroid linkage model: 49 biomarker genes.

	DS vs. LS	LH vs. LR	DS_LS vs. LH_LR	DR vs. DS_LS_LH_LR	DH vs. DS_LS_LH_LR_DR
Genes	CA334854 TNFRSF6B AX746743 BU189136 PLEKHB2 IL1RAPL2 DUSP21 NUFIP2 LIPJ ARSK MAF1 PRKD2	C15orf5 MANBA VAX2	FAM108B1 DR731427 BE646122 ROBO1 ZNF234 DSCAM MARK2 ZFP91 METTL6 C1orf147	METTL2A FAM170B BG944228 PDCD7 FBXO41 ZNF121 TRPC5 ASAP1 ASXL1 ATL1 FOSB WIPI2 BF594823 AL71228	FAM114A1 CDC42BPG C14orf145 ANKLE1 OR2G6 UPF3B MGA FGF16 AKIP1 AA884297

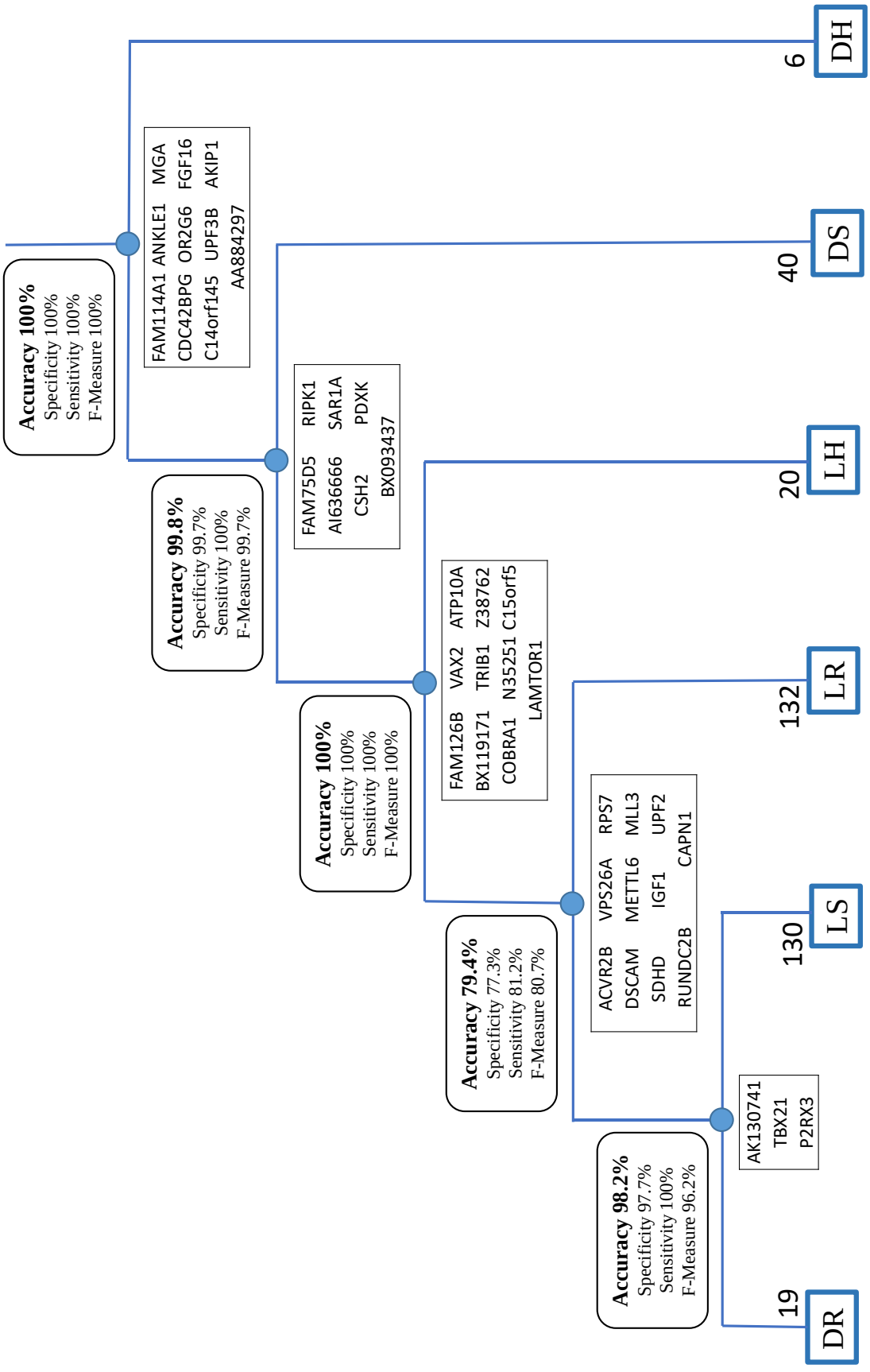

Figure 60. Single linkage model: classification model with performance measures.

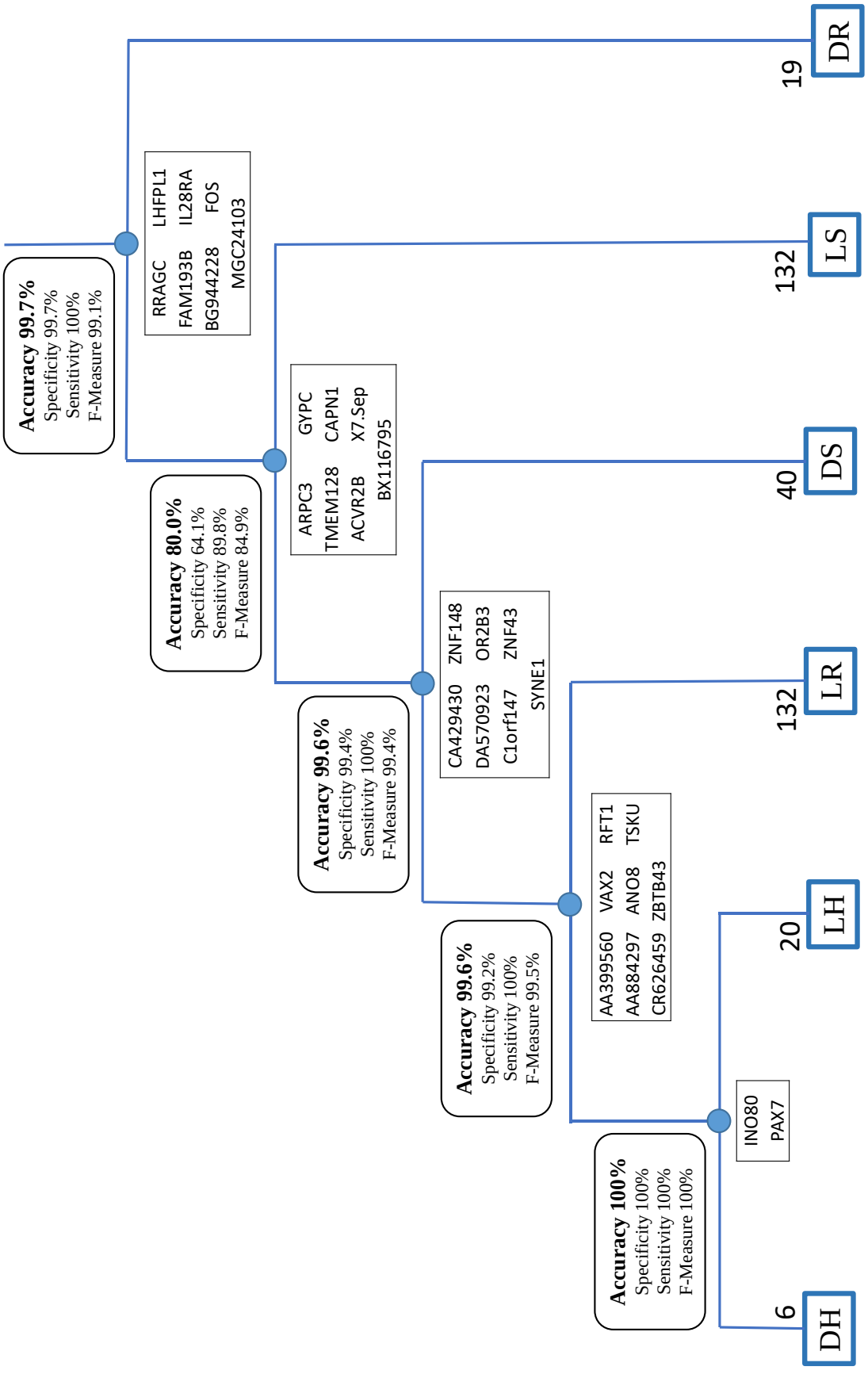

Figure 61. Complete linkage model: classification model with performance measures.

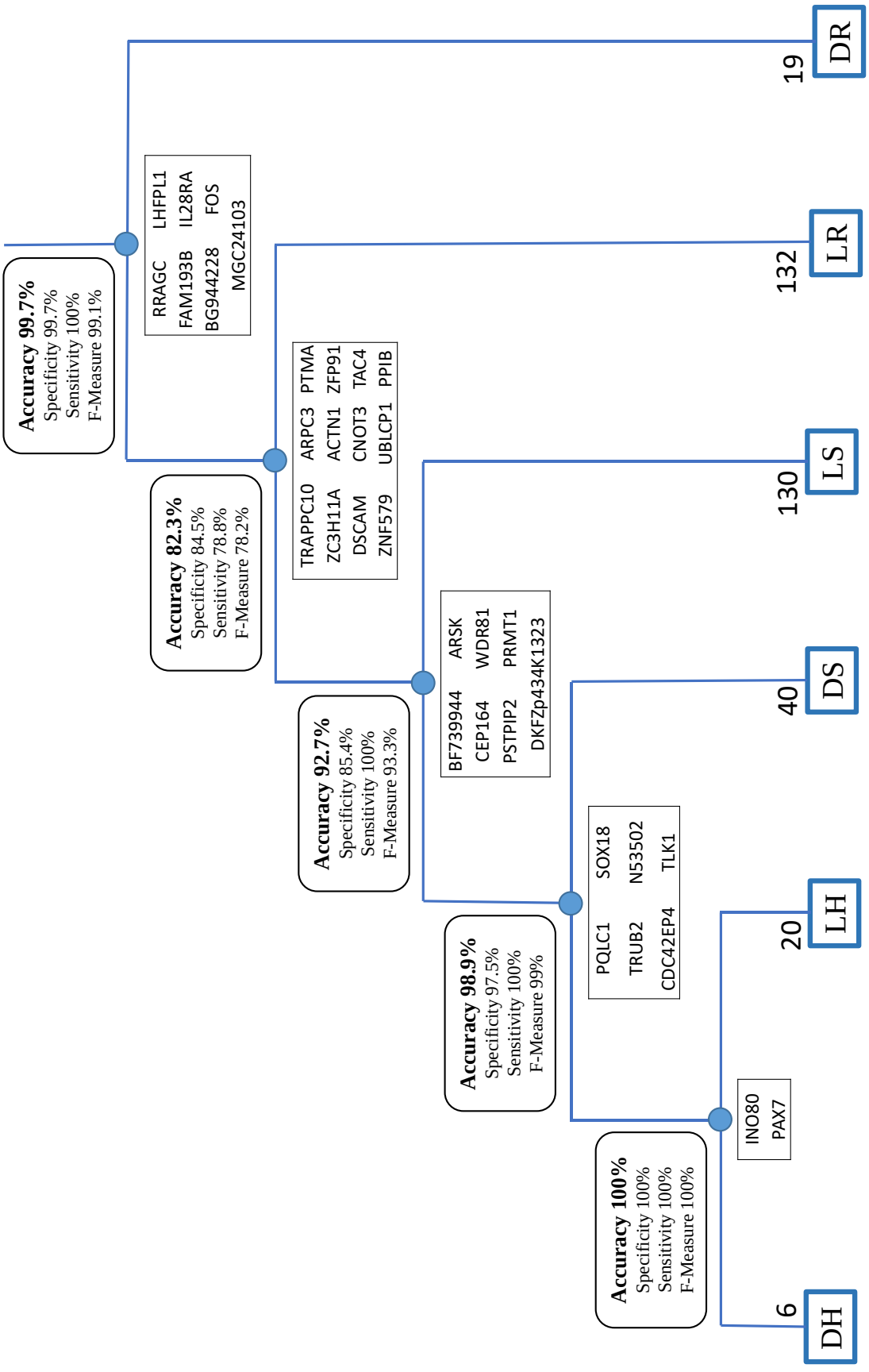

Figure 62. Average linkage model: classification model with performance measures.

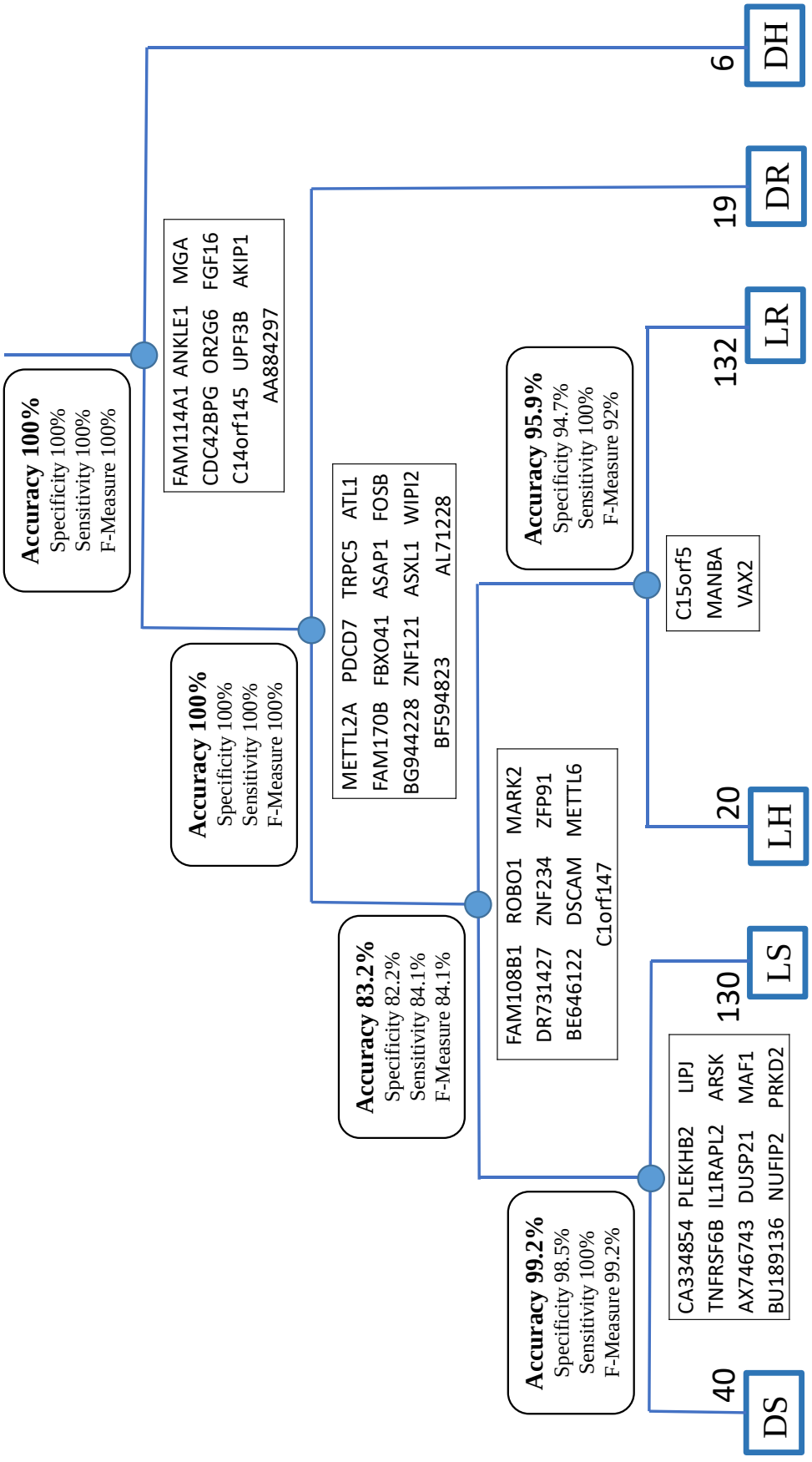

Figure 63. Centroid linkage model: The classification model with performance measures.

CHAPTER 6. CONCLUSION

6.1 Overview

In this chapter, the novelties and contributions of this work are highlighted. The significance and benefits are also discussed. The research plan and timeline are depicted. Then future work and the final conclusions are discussed at the end.

6.2 Novelties and Contributions

The novelties and contributions of this work include the following:

6.2.1 Manufacturing Domain

- The creation of an integrative model for modern systems that consists of big data, IoT, CPS, machine learning, and deep learning. These components are represented in a state-of-the-art model for a real-time decision-making process.
- Adding the missing enabler (Big data analytics) that iFactory lacks to convert it for Industry 4.0.
- Deep leaning convolutional neural network, which is optimized by ADAM optimizer and regularized by drop-out approach for the internal nodes in the network, is introduced in the manufacturing domain for the iFactory product. The network was trained and refined by adding many layers using a greedy method to obtain the best structure that balances high-performance measurements without overfitting. For the best of our knowledge, no one in the literature has applied the deep learning CNN using big data with a closed cycle in the manufacturing system.
- The extraction of hidden features in the iFactory product using convolution layers in deep learning, which can be used to predict unusual situations, such as the production of defective products. The extracted features are visual features that can distinguish the fine looking (normal) product from the defected product. This decisive feature speeds up prediction performance by reducing the dimensionality of the learning data.

6.2.2 Health Informatics Domain

- As a health informatics contribution, a feature selection and supervised learning model was developed to identify gene biomarkers that are related to breast cancer survivability based on specific treatments. To the best of our knowledge, this is the first model that can guide survivability based on treatment.
- The supervised model in the health informatics domain consists of state-of-art combination methods of feature filtering using different rankers such as Information Gain and ChiSquare, wrapper feature selection method which is mRMR, and standard classifiers that greedily

optimized such as SVM and random forest. This combination of methods was structured hierarchically in tree nodes one versus rest model. The novelty of this unique combination has never been introduced in the literature, to the best of our knowledge.

- The development of a hierarchical one-versus-rest model for predicting five-year survival based on the treatment and gene expression. Studying the genomic activities of five-year breast cancer survivors can provide the strength factors of resisting cancer in the body. The pulled-up genes from the computational model were validated using the literature, which confirms the findings.
- The accuracy of the classification model in a healthcare application achieved high measurment compared to current classification literature for the same types of health outcomes.
- The proposed prediction models in the health informatics domain outperformed other state-of-the-art prediction models in the field. Various performance measurements are calculated such as sensitivity, specificity, and F-measure, as presented in chapter 4 for the manufacturing domain and in chapter 5 for health informatics domain.
- Techniques to overcome Class-Imbalance (C-I) were presented to handle what is known as the curse of dimensionality problem, where the number of samples is much smaller than the number of features.
- As a machine learning contribution, the use of the semi-supervised learning model is proposed to identify the borders between classes in the multi-class learning problem.

6.3 Research Significance and Benefits

6.3.1 Manufacturing Domain

This research will contribute to the development of a key enabler for Industry 4.0, a big data decision-making tool, as a modern global trend in sustainable manufacturing. The proposed method can be used to optimize manufacturing processes in a wide variety of industries. The successful implementation of this research in manufacturing facilities will lead to not only cost savings in these facilities but also the achievement of feasible sustainable systems.

- The proposed model was designed and developed for gathering, analyzing and processing big data generated from complex manufacturing systems (structured or non-structured).
- The model supports real-time decision-making and knowledge discovery using big data analytics and data mining methods.
- The model is a key enabler for Smart Manufacturing (Industry 4.0) as a modern global trend in sustainable manufacturing.
- Defects detection model using CNN.
- Will not only lead to cost saving in manufacturing facilities but will also lead to the achievement of feasible, sustainable systems.

6.3.2 Health Informatics Domain

The proposed methods are applied in the health informatics field, and modern systems are shown to be effective for the diagnosis and the treatment of cancer. Cancer survivability genes were pulled-up and proven to be related to disease development. The model can personalize treatment based on the patient's genome.

- Supervised ML and un/semi-supervised ML models with feature selection were developed to identify gene biomarkers that are related to breast cancer survivability based on specific treatments.
- The work identified 47/37 gene biomarkers; a functional validation is done for genes by studying the gene cards. Some genes were found related to cancer in previous studies, while others are the new finding biomarkers that need more study and analysis.
- The results show that a few numbers of gene biomarkers (gene signature) at each node that can determine the class with accuracy around 99% for survival living / deceased based on treatments which are vital to ensure that the patients will have the best potential response to a specific therapy. These signatures will be used as a predictor of survival in breast cancer.

6.4 Limitations

The main contribution of the manufacturing domain is converting the i3.0 into i4.0. However, the conversion process might not be straight forward to cover the whole manufacturing process. In addition to the lack of CPS technologies in some industrial areas, some challenges such as locating the sensors and cameras around the production line might not be partial or fully applicable in the whole manufacturing process. The infrastructure requirements are costly and might not be available in some factories. The availability of real data is hard to be found in manufacturing system domain, due to copyrights and privacy concerns from the owners who are reluctant to share the data publically.

Since this area is new, the maintenance costs and plans are not clear and needed to be feasibly studied. The downtimes of different scenarios are not covered in this thesis. To convert the proposed method into commercial version. A well planned business models have to be analyzed before start implementing the changes. One last thing to consider that each manufacturing system has it is own properties, which means that the system should be profiled and studied well before adopting the proposed changes.

For the health domain shortcomings, the potential pulled biomarker genes have to undergo wet-lab experiments to study the relevancy with cancer. These genes will interact with the other genes and drugs, a team from medical, pharmaceutical, biological domain have to team up to analyze the new found none reported to cancer genes before considering them in the diagnosis and treatment guidance.

6.5 Future Work

In the manufacturing domain, CNN is applied to product images in the core model in the learning phase. The iFactory sensors generate different types of data, such as images, log files, and other signals. Text log files can potentially be analyzed using natural language processing (NLP) and fed into a learning module. NLP may add better modularity if it utilizes communication data among the iFactory modules in the learning and prediction phases. Signals from thermal sensors can be processed using signal processing techniques and added as an input layer to the CNN, which increase the learning domain of the module. The iFactory model is connected to the server as one unit. The model can be expanded to connect multiple iFactories and add manufacturing management systems to handle requests from multiple industrial sites.

The bioengineering modules analyzed the biomarker gene expressions to predict survivability. The dataset has multiple sources of biological data, such as copy number alterations (CNAs), gene mutations, and insertion and deletion events (indels). These sources are known as multi-omics data. The proposed models can be improved by utilizing the multi-omics data in the learning and predictions phases. Genomics indels and mutations cause different body response to medication. Utilizing multi-omics data may provide a comprehensive understanding of disease progression.

Some important future work is to visualize the stored data, learning process, and the taken decisions in useful charts and dashboards. Data visualization is a hot topic nowadays, and its key strength is its ability to communicate through visual arts directly. The trends of the data regarding time can also be analyzed through time-series analysis.

This novel model can be improved for use in the identification of proper biomarker genes (signature) for different cancer types or even in cases in which patients need or have received more than one type of therapy. Considering additional patient data will enable researchers to cover all missing treatments. With this considerable data size, big data tools such as Hadoop and Spark can be utilized to devise an enhanced model.

6.6 Conclusion

The main contribution of this thesis is the creation of an integrative modern industrial system that utilizes big data, machine learning, deep learning, the IoT, and super-computers for online decision making. These decisions vary from monitoring and detecting unusual situations and extracting hidden decisive features to predicting labels or the type of the instance (e.g., a breast cancer patient's subtype or defected products). We applied the proposed methods in two domains, manufacturing systems and bioengineering.

In the manufacturing domain, the iFactory system was enhanced to add all Industry 4.0 enablers. The big data that is continuously generated by iFactory sensors and cameras were connected to the Compute-Canada clusters where the learning system is hosted. A CNN deep learning approach was trained using previously-generated data, who receives the data (instance) online, can take the decision for the newer data, and update the learning system from the newer instances. The learning system can detect the most decisive hidden features of the instances (the product in this case). The decision will be made online, so the iFactory can react based on the decision. A full cycle of the deep learning CNN on big data for real-time decision is introduced as a closed cycle. The results show high performance with 97% accuracy to distinguish both classes, which are a normal product and the defective products.

In the second domain, the breast cancer data set (METABRIC) was analyzed to predict the survivability and the best treatment for the patient. The model was built using big data analysis and machine learning techniques to make health-outcome predictions. The model can make an online prediction based on the genomic activity of the patient. The learning-system-extracted biomarker genes can be used to predict outcomes using gene expression data decisively. These supervised models can assist in the diagnosis and treatment of people with the disease and enhance the patient's quality of life by reducing treatment trials by selecting the most effective treatment. The Tree model shows a high-performance measurements start with an accuracy of 100% for the first node where the DH class is distinguished from the rest. The specificity of the classification at this node is 100%, the sensitivity and F-measure both are 100%. At the second node where DR class is distinguished from the rest , all the performance measurements have 100%. The performance measurements at the third node where the LH class is distinguished from the rest, also have 100%. While in the fourth node where the DS class is distinguished from the rest, The accuracy is 97.9%, the specificity is 100%, sensitivity is 96.9%, and the F-measure is 98.5%. Finally, in the last node where the model distinguishes the remaining two class LR and LS from each other, the accuracy is 80.9%, the specificity is 76.9%, the sensitivity is 84.8%, and the F-measure is 81.8%.

REFERENCES

Abou Tabl, A., A. Alkhateeb, W. ElMaraghy and A. Ngom (2017). Machine learning model for identifying gene biomarkers for breast cancer treatment survival. Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, ACM.

Abou Tabl, A. and W. ElMaraghy (2019). Big Data Analytics for Defects Detection in Manufacturing Systems (Industry 4.0). International Journal of Computer Integrated Manufacturing(IJCIM) ID: TCIM-2019-IJCIM-0178. , IJCIM.

Allegra, A., A. Alonci, S. Campo, G. Penna, A. Petrungaro, D. Gerace and C. Musolino (2012). "Circulating microRNAs: new biomarkers in diagnosis, prognosis and treatment of cancer." International journal of oncology 41(6): 1897-1912.

Auschitzky, E., M. Hammer and A. Rajagopaul (2014). "How big data can improve manufacturing." McKinsey & Company.

Babiceanu, R. F. and R. Seker (2016). "Big Data and virtualization for manufacturing cyber-physical systems: A survey of the current status and future outlook." Computers in Industry 81: 128-137.

Baheti, R. and H. Gill (2011). "Cyber-physical systems." The impact of control technology 12: 161-166.

Bamberger, A. M., C. Methner, B. W. Lisboa, C. Städtler, H. M. Schulte, T. Löning and K. Milde-Langosch (1999). "Expression pattern of the AP-1 family in breast cancer: Association of fosB expression with a well-differentiated, receptor-positive tumor phenotype." International journal of cancer 84(5): 533-538.

Belle, A., R. Thiagarajan, S. Soroushmehr, F. Navidi, D. A. Beard and K. Najarian (2015). "Big data analytics in healthcare." BioMed research international 2015.

Bengio, Y. (2009). "Learning deep architectures for AI." Foundations and trends® in Machine Learning 2(1): 1-127.

Berman, J. J. (2013). Principles of big data: preparing, sharing, and analyzing complex information, Newnes.

Bhatia, N. (2010). "Survey of nearest neighbor techniques." arXiv preprint arXiv:1007.0085.

Biostars. (2019). "Human genes, transcripts, and proteins (Visited Feb 10, 2019)." Retrieved Feb, 2019, 2019, from https://www.biostars.org/p/241563/.

Bishop, B. (2006). "CM: Pattern Recognition and Machine Learning." Journal of Electronic Imaging 16(4): 140-155.

Boltzmann, L. (1868). "Studien uber das Gleichgewicht der lebenden Kraft." Wissenschafiliche Abhandlungen 1: 49-96.

Borgia, E. (2014). "The Internet of Things vision: Key features, applications and open issues." Computer Communications 54: 1-31.

Breiman, L. (2001). "Random forests." Machine learning 45(1): 5-32.

Caballero, O. L., S. Shousha, Q. Zhao, A. J. Simpson, R. C. Coombes and A. M. Neville (2014). "Expression of Cancer/Testis genes in ductal carcinoma in situ and benign lesions of the breast." Oncoscience 1(1): 14.

Cai, Q., B. Zhang, H. Sung, S.-K. Low, S.-S. Kweon, W. Lu, J. Shi, J. Long, W. Wen and J.-Y. Choi (2014). "Genome-wide association analysis in East Asians identifies breast cancer susceptibility loci at 1q32. 1, 5q14. 3 and 15q26. 1." Nature genetics 46(8): 886.

Cardoso, F., L. J. van't Veer, J. Bogaerts, L. Slaets, G. Viale, S. Delaloge, J.-Y. Pierga, E. Brain, S. Causeret and M. DeLorenzi (2016). "70-gene signature as an aid to treatment decisions in early-stage breast cancer." New England Journal of Medicine 375(8): 717-729.

Cerami, E., J. Gao, U. Dogrusoz, B. E. Gross, S. O. Sumer, B. A. Aksoy, A. Jacobsen, C. J. Byrne, M. L. Heuer and E. Larsson (2012). The cBio cancer genomics portal: an open platform for exploring multidimensional cancer genomics data, AACR.

Chang, C.-C. (2011). "" LIBSVM: a library for support vector machines," ACM Transactions on Intelligent Systems and Technology, 2: 27: 1--27: 27, 2011." http://www. csie. ntu. edu. tw/~ cjlin/libsvm 2.

Chang, H. Y., D. S. Nuyten, J. B. Sneddon, T. Hastie, R. Tibshirani, T. Sørlie, H. Dai, Y. D. He, L. J. van't Veer and H. Bartelink (2005). "Robustness, scalability, and integration of a wound-response gene expression signature in predicting breast cancer survival." Proceedings of the National Academy of Sciences 102(10): 3738-3743.

Chawla, N. V., K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer (2002). "SMOTE: synthetic minority over-sampling technique." Journal of artificial intelligence research 16: 321-357.

Chen, H. (2017). "Applications of cyber-physical system: a literature review." Journal of Industrial Integration and Management 2(03): 1750012.

Cheng, L. and T. Yu (2018). "Dissolved gas analysis principle-based intelligent approaches to fault diagnosis and decision making for large oil-immersed power transformers: A survey." Energies 11(4): 913.

Cheng, L. and T. Yu (2019). "A new generation of AI: A review and perspective on machine learning technologies applied to smart energy and electric power systems." International Journal of Energy Research.

Chiaretti, S., X. Li, R. Gentleman, A. Vitale, M. Vignetti, F. Mandelli, J. Ritz and R. Foa (2004). "Gene expression profile of adult T-cell acute lymphocytic leukemia identifies distinct subsets of patients with different response to therapy and survival." Blood 103(7): 2771-2778.

Choudhary, A. K., J. A. Harding and M. K. Tiwari (2009). "Data mining in manufacturing: a review based on the kind of knowledge." Journal of Intelligent Manufacturing 20(5): 501.

Croft, D., A. F. Mundo, R. Haw, M. Milacic, J. Weiser, G. Wu, M. Caudy, P. Garapati, M. Gillespie and M. R. Kamdar (2013). "The Reactome pathway knowledgebase." Nucleic acids research 42(D1): D472-D477.

Curtis, C., S. P. Shah, S.-F. Chin, G. Turashvili, O. M. Rueda, M. J. Dunning, D. Speed, A. G. Lynch, S. Samarajiwa and Y. Yuan (2012). "The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups." Nature 486(7403): 346.

DeGregory, K., P. Kuiper, T. DeSilvio, J. Pleuss, R. Miller, J. Roginski, C. Fisher, D. Harness, S. Viswanath and S. Heymsfield (2018). "A review of machine learning in obesity." Obesity Reviews 19(5): 668-685.

Dimitrov, D. V. (2016). "Medical internet of things and big data in healthcare." Healthcare informatics research 22(3): 156-163.

Dombkowski, A. A., Z. Sultana, D. B. Craig and H. Jamil (2011). "In silico analysis of combinatorial microRNA activity reveals target genes and pathways associated with breast cancer metastasis." Cancer informatics 10: CIN. S6631.

Domingos, P. and M. Pazzani (1997). "On the optimality of the simple Bayesian classifier under zero-one loss." Machine learning 29(2-3): 103-130.

Dubey, R., A. Gunasekaran, S. J. Childe, S. F. Wamba and T. Papadopoulos (2016). "The impact of big data on world-class sustainable manufacturing." The International Journal of Advanced Manufacturing Technology 84(1-4): 631-645.

Dumbill, E. (2013). "A Revolution That Will Transform How We Live, Work, and Think: An Interview with the Authors of Big Data." Big Data 1(2): 73-77.

Dutta, D. and I. Bose (2015). "Managing a big data project: the case of ramco cements limited." International Journal of Production Economics 165: 293-306.

EigenInovations. (2019). "Making Complex Quality Inspection Simple (Visited Jan 20, 2019)." from http://eigeninnovations.com/.

Elkarami, B., A. Alkhateeb and L. Rueda (2016). Cost-sensitive classification on class-balanced ensembles for imbalanced non-coding RNA data. 2016 IEEE EMBS International Student Conference (ISC), IEEE.

ElMaraghy, H., T. AlGeddawy, A. Azab and W. ElMaraghy (2012). Change in manufacturing–research and industrial challenges. Enabling manufacturing competitiveness and economic sustainability, Springer: 2-9.

ElMaraghy, H. A. (2005). "Flexible and reconfigurable manufacturing systems paradigms." International journal of flexible manufacturing systems 17(4): 261-276.

ElMaraghy, W. (2009). "Knowledge Management in collaborative engineering." International Journal of Collaborative Engineering 1(1-2): 114-124.

Fang, R., S. Pouyanfar, Y. Yang, S.-C. Chen and S. Iyengar (2016). "Computational health informatics in the big data age: a survey." ACM Computing Surveys (CSUR) 49(1): 12.

Fayyad, U. and P. Stolorz (1997). "Data mining and KDD: Promise and challenges." Future generation computer systems 13(2-3): 99-115.

Frenkel-Morgenstern, M., A. Gorohovski, S. Tagore, V. Sekar, M. Vazquez and A. Valencia (2017). "ChiPPI: a novel method for mapping chimeric protein–protein interactions uncovers selection principles of protein fusion events in cancer." Nucleic acids research 45(12): 7094-7105.

Glawara, R., Z. Kemenyc, T. Nemetha, K. Matyasb, L. Monostoric and W. Sihna (2015). "A holistic approach for quality oriented maintenance planning supported by data mining methods."

Gölzer, P., P. Cato and M. Amberg (2015). Data Processing Requirements of Industry 4.0-Use Cases for Big Data Applications. Proceedings of the European Conference on Information Systems (ECIS) 2015.

Goodfellow, I., Y. Bengio, A. Courville and Y. Bengio (2016). Deep learning, MIT press Cambridge.

Gošev, I., M. Zeljko, Ž. Đurić, I. Nikolić, M. Gošev, S. Ivčević, D. Bešić, Z. Legčević and F. Paić (2017). "Epigenome alterations in aortic valve stenosis and its related left ventricular hypertrophy." Clinical epigenetics 9(1): 106.

Gross, S. (1980). Median estimation in sample surveys. Proceedings of the Section on Survey Research Methods, American Statistical Association Alexandia, VA.

Gu, F., M. S. Doderer, Y.-W. Huang, J. C. Roa, P. J. Goodfellow, E. L. Kizer, T. H. Huang and Y. Chen (2013). "CMS: a web-based system for visualization and analysis of genome-wide methylation data of human cancers." PloS one 8(4): e60980.

Hahnloser, R. H. R., R. Sarpeshkar, M. A. Mahowald, R. J. Douglas and H. S. Seung (2000). "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit." Nature 405: 947.

Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten (2009). "The WEKA data mining software: an update." ACM SIGKDD explorations newsletter 11(1): 10-18.

Hallonet, M., T. Hollemann, R. Wehr, N. A. Jenkins, N. G. Copeland, T. Pieler and P. Gruss (1998). "Vax1 is a novel homeobox-containing gene expressed in the developing anterior ventral forebrain." Development 125(14): 2599-2610.

Hazen, B. T., C. A. Boone, J. D. Ezell and L. A. Jones-Farmer (2014). "Data quality for data science, predictive analytics, and big data in supply chain management: An introduction to the problem and suggestions for research and applications." International Journal of Production Economics 154: 72-80.

Hinton, G. E. and R. R. Salakhutdinov (2006). "Reducing the dimensionality of data with neural networks." science 313(5786): 504-507.

Hopfield, J. J. (1982). "Neural networks and physical systems with emergent collective computational abilities." Proceedings of the national academy of sciences 79(8): 2554-2558.

IBM. (2019). "(Visual Insights) Resolve quality problems on the factory

floor faster with artificial intelligence (Visited March 26, 2019)." Retrieved March, 2019, 2019, from https://www.ibm.com/internet-of-things/ae-en/iot-solutions/iot-manufacturing/.

Jermyjordan. (2019). "convolution neural network (CNN) layers (Visited Feb 10, 2019)." Retrieved Feb, 2019, 2019, from https://www.jeremyjordan.me/convolutional-neural-networks/.

Johnson, S. C. (1967). "Hierarchical clustering schemes." Psychometrika 32(3): 241-254.

Joseph Bradley, X. M., and Denny Lee. (2016). "Why you should use Spark for machine learning." from http://www.infoworld.com/article/3031690/analytics/why-you-should-use-spark-for-machine-learning.html.

JU, P., X. Zhou and W. Chen (2018). "Smart grid plus research overview." Electric Power Automation Equipment, Nanning 38(5): 2-11.

kaggle. (2019). "ResNet-50." Retrieved Jan 2019, 2019, from https://www.kaggle.com/keras/resnet50.

Kashkoush, M. and H. ElMaraghy (2015). "Knowledge-based model for constructing master assembly sequence." Journal of Manufacturing Systems 34: 43-52.

Katoh, M. and H. Nakagama (2014). "FGF receptors: cancer biology and therapeutics." Medicinal research reviews 34(2): 280-300.

Kechavarzi, B. and S. C. Janga (2014). "Dissecting the expression landscape of RNA-binding proteins in human cancers." Genome biology 15(1): R14.

Kharlamov, A. and V. Podlozhnyuk (2007). "Image denoising." NVIDIA.

Kim, Y. (2014). "Convolutional neural networks for sentence classification." arXiv preprint arXiv:1408.5882.

Kingma, D. P. and J. Ba (2014). "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980.

Koren, Y. and M. Shpitalni (2010). "Design of reconfigurable manufacturing systems." Journal of manufacturing systems 29(4): 130-141.

Krizhevsky, A., I. Sutskever and G. E. Hinton (2012). Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems.

Kumaraguru, S. and K. Morris (2014). Integrating real-time analytics and continuous performance management in smart manufacturing systems. IFIP International Conference on Advances in Production Management Systems, Springer.

Labhart, P., S. Karmakar, E. M. Salicru, B. S. Egan, V. Alexiadis, B. W. O'Malley and C. L. Smith (2005). "Identification of target genes in breast cancer cells directly regulated by the SRC-3/AIB1 coactivator." Proceedings of the National Academy of Sciences 102(5): 1339-1344.

LeCun, Y., B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard and L. D. Jackel (1990). Handwritten digit recognition with a back-propagation network. Advances in neural information processing systems.

Lee, E. A. (2008). Cyber physical systems: Design challenges. 11th IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC), IEEE.

Lee, J., B. Bagheri and H.-A. Kao (2014). Recent advances and trends of cyber-physical systems and big data analytics in industrial informatics. International Conference on Industrial Informatics (INDIN).

Lee, J., E. Lapira, B. Bagheri and H.-a. Kao (2013). "Recent advances and trends in predictive manufacturing systems in big data environment." Manufacturing Letters 1(1): 38-41.

Li, B.-H., L. Zhang, S.-L. Wang, F. Tao, J. Cao, X. Jiang, X. Song and X. Chai (2010). "Cloud manufacturing: a new service-oriented networked manufacturing model." Computer integrated manufacturing systems 16(1): 1-7.

Li, J., F. Tao, Y. Cheng and L. Zhao (2015). "Big Data in product lifecycle management." The International Journal of Advanced Manufacturing Technology 81(1-4): 667-684.

Li, S., L. Da Xu and S. Zhao (2018). "5G internet of things: A survey." Journal of Industrial Information Integration.

Li, X., D. Li, J. Wan, A. V. Vasilakos, C.-F. Lai and S. Wang (2017). "A review of industrial wireless networks in the context of industry 4.0." Wireless networks 23(1): 23-41.

Li, Y., X. Liu, H. Tang, H. Yang and X. Meng (2017). "RNA Sequencing Uncovers Molecular Mechanisms Underlying Pathological Complete Response to Chemotherapy in Patients with Operable Breast Cancer." Medical science monitor: international medical journal of experimental and clinical research 23: 4321.

Li, Y., C. K. Tsang, S. Wang, X. X. Li, Y. Yang, L. Fu, W. Huang, M. Li, H. Y. Wang and X. S. Zheng (2016). "MAF1 suppresses AKT-mTOR signaling and liver cancer through activation of PTEN transcription." Hepatology 63(6): 1928-1942.

Liu, R., B. Yang, E. Zio and X. Chen (2018). "Artificial intelligence for fault diagnosis of rotating machinery: A review." Mechanical Systems and Signal Processing 108: 33-47.

Liu, W., Z. Wang, X. Liu, N. Zeng, Y. Liu and F. E. Alsaadi (2017). "A survey of deep neural network architectures and their applications." Neurocomputing 234: 11-26.

Loshin, D. (2013). Big data analytics: from strategic planning to enterprise integration with tools, techniques, NoSQL, and graph, Elsevier.

Lu, Y. (2017). "Cyber physical system (CPS)-based industry 4.0: a survey." Journal of Industrial Integration and Management 2(03): 1750014.

Lu, Y. (2017). "Industry 4.0: A survey on technologies, applications and open research issues." Journal of Industrial Information Integration 6: 1-10.

Lv, C., X. Hu, A. Sangiovanni-Vincentelli, Y. Li, C. M. Martinez and D. Cao (2019). "Driving-Style-Based Codesign Optimization of an Automated Electric Vehicle: A Cyber-Physical System Approach." IEEE Transactions on Industrial Electronics 66(4): 2965-2975.

Mangasarian, Y.-J. and W. Wolberg (2000). Breast cancer survival and chemotherapy: a support vector machine analysis. Discret Math Probl with Med Appl DIMACS Work Discret Math Probl with Med Appl December 8–10, 1999, Volume 55: 1.

Mantel, N. (1963). "Chi-square tests with one degree of freedom; extensions of the Mantel-Haenszel procedure." Journal of the American Statistical Association 58(303): 690-700.

Manyika, J., M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh and A. H. Byers (2011). "Big data: The next frontier for innovation, competition, and productivity."

Mashal, I., O. Alsaryrah, T.-Y. Chung, C.-Z. Yang, W.-H. Kuo and D. P. Agrawal (2015). "Choices for interaction with things on Internet and underlying issues." Ad Hoc Networks 28: 68-90.

Mendiratta, S., S. Bhatia, S. Jain, T. Kaur and V. Brahmachari (2016). "Interaction of the chromatin remodeling protein hINO80 with DNA." PloS one 11(7): e0159370.

Mikusz, M. (2014). "Towards an understanding of cyber-physical systems as industrial software-product-service systems." Procedia CIRP 16: 385-389.

Miller, K. D., R. L. Siegel, C. C. Lin, A. B. Mariotto, J. L. Kramer, J. H. Rowland, K. D. Stein, R. Alteri and A. Jemal (2016). "Cancer treatment and survivorship statistics, 2016." CA: a cancer journal for clinicians 66(4): 271-289.

Minn, A. J., G. P. Gupta, P. M. Siegel, P. D. Bos, W. Shu, D. D. Giri, A. Viale, A. B. Olshen, W. L. Gerald and J. Massagué (2005). "Genes that mediate breast cancer metastasis to lung." Nature 436(7050): 518.

Mitchell, T. (2005). Machine learning. Chapter generative and discriminative classifiers: naive Bayes and logistic regression.

Moghaddam, M. and S. Y. Nof (2018). "Collaborative service-component integration in cloud manufacturing." International Journal of Production Research 56(1-2): 677-691.

Mourtzis, D. and M. Doukas (2014). "Design and planning of manufacturing networks for mass customisation and personalisation: challenges and outlook." Procedia CIRP 19: 1-13.

Moyne, J. and J. Iskandar (2017). "Big data analytics for smart manufacturing: Case studies in semiconductor manufacturing." Processes 5(3): 39.

Nam, S., H. R. Chang, H. R. Jung, Y. Gim, N. Y. Kim, R. Grailhe, H. R. Seo, H. S. Park, C. Balch and J. Lee (2015). "A pathway-based approach for identifying biomarkers of tumor progression to trastuzumab-resistant breast cancer." Cancer letters 356(2): 880-890.

Narayanan, A. N., R. Ak, Y.-T. T. Lee, R. Ghosh and S. Rachuri (2017). Summary of the symposium on data analytics for advanced manufacturing.

Network, C. G. A. (2012). "Comprehensive molecular portraits of human breast tumours." Nature 490(7418): 61.

NIST, N. I. o. S. a. T. (2018). "Data Analytics for Smart Manufacturing Systems."

NSF. (2018). "Cyber-Physical Systems (CPS)." from https://www.nsf.gov/pubs/2017/nsf17529/nsf17529.pdf.

Núñez, M. (1988). Economic Induction: A Case Study. EWSL.

Obitko, M., V. Jirkovský and J. Bezdíček (2013). Big data challenges in industrial automation. International Conference on Industrial Applications of Holonic and Multi-Agent Systems, Springer.

Paredes-Aracil, E., A. Palazón-Bru, D. M. Folgado-de la Rosa, J. R. Ots-Gutiérrez, A. F. Compañ-Rosique and V. F. Gil-Guillén (2017). "A scoring system to predict breast cancer mortality at 5 and 10 years." Scientific reports 7(1): 415.

Paschke, L., M. Rucinski, A. Ziolkowska, T. Zemleduch, W. Malendowicz, Z. Kwias and L. K. Malendowicz (2014). "ZFP91—a newly described gene potentially involved in prostate pathology." Pathology & Oncology Research 20(2): 453-459.

Pederson, H. J., N. Hussain, R. Noss, C. Yanda, C. O'Rourke, C. Eng and S. R. Grobmyer (2018). "Impact of an embedded genetic counselor on breast cancer treatment." Breast cancer research and treatment: 1-4.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss and V. Dubourg (2011). "Scikit-learn: Machine learning in Python." Journal of machine learning research 12(Oct): 2825-2830.

Peng, H., F. Long and C. Ding (2005). "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy." IEEE Transactions on pattern analysis and machine intelligence 27(8): 1226-1238.

Pereira, B., S.-F. Chin, O. M. Rueda, H.-K. M. Vollan, E. Provenzano, H. A. Bardwell, M. Pugh, L. Jones, R. Russell and S.-J. Sammut (2016). "The somatic mutation profiles of 2,433 breast cancers refine their genomic and transcriptomic landscapes." Nature communications 7: 11479.

Perou, C. M., T. Sørlie, M. B. Eisen, M. Van De Rijn, S. S. Jeffrey, C. A. Rees, J. R. Pollack, D. T. Ross, H. Johnsen and L. A. Akslen (2000). "Molecular portraits of human breast tumours." nature 406(6797): 747.

Peruzzini, M., F. Gregori, A. Luzi, M. Mengarelli and M. Germani (2017). "A social life cycle assessment methodology for smart manufacturing: The case of study of a kitchen sink." Journal of Industrial Information Integration 7: 24-32.

Prat, A. and C. M. Perou (2011). "Deconstructing the molecular portraits of breast cancer." Molecular oncology 5(1): 5-23.

Qiu, X., H. Luo, G. Xu, R. Zhong and G. Q. Huang (2015). "Physical assets and service sharing for IoT-enabled Supply Hub in Industrial Park (SHIP)." International Journal of Production Economics 159: 4-15.

Qiu, Z., W. Guo, Q. Wang, Z. Chen, S. Huang, F. Zhao, M. Yao, Y. Zhao and X. He (2015). "MicroRNA-124 reduces the pentose phosphate pathway and proliferation by targeting PRPS1 and RPIA mRNAs in human colorectal cancer cells." Gastroenterology 149(6): 1587-1598. e1511.

Roy, U., B. Zhu, Y. Li, H. Zhang and O. Yaman (2014). Mining Big Data in Manufacturing: Requirement Analysis, Tools and Techniques. ASME 2014 International Mechanical Engineering Congress and Exposition, American Society of Mechanical Engineers.

Sabe, H., S. Hashimoto, M. Morishige, E. Ogawa, A. Hashimoto, J. M. Nam, K. Miura, H. Yano and Y. Onodera (2009). "The EGFR-GEP100-Arf6-AMAP1 signaling pathway specific to breast cancer invasion and metastasis." Traffic 10(8): 982-993.

Santos, M. Y., B. Martinho and C. Costa (2017). "Modelling and implementing big data warehouses for decision support." Journal of Management Analytics 4(2): 111-129.

Schmidhuber, J. (2015). "Deep learning in neural networks: An overview." Neural networks 61: 85-117.

Schölkopf, B., A. J. Smola and F. Bach (2002). Learning with kernels: support vector machines, regularization, optimization, and beyond, MIT press.

Steve Chadwick, D. L., Steven J. Meyer, Joe Sartini. (2016). "Intel White Paper: Using Big Data in Manufacturing at Intel's Smart Factories." Retrieved March, 2019, 2019, from https://www.intel.com/content/dam/www/public/us/en/documents/best-practices/using-big-data-in-manufacturing-at-intels-smart-factories-paper.pdf.

Stuhlmiller, T. J., S. M. Miller, J. S. Zawistowski, K. Nakamura, A. S. Beltran, J. S. Duncan, S. P. Angus, K. A. Collins, D. A. Granger and R. A. Reuther (2015). "Inhibition of lapatinib-induced kinome reprogramming in ERBB2-positive breast cancer by targeting BET family bromodomains." Cell reports 11(3): 390-404.

Sun, C. (2012). "Application of RFID technology for logistics on internet of things." AASRI Procedia 1: 106-111.

Tabl, A. A., A. Alkhateeb, W. ElMaraghy, L. Rueda and A. Ngom (2019). "A machine learning approach for identifying gene biomarkers guiding the treatment of breast cancer." Frontiers in Genetics 10.

Tabl, A. A., A. Alkhateeb, H. Q. Pham, L. Rueda, W. ElMaraghy and A. Ngom (2018). "A novel approach for identifying relevant genes for breast cancer survivability on specific therapies." Evolutionary Bioinformatics 14: 1176934318790266.

Tabl, A. A., A. Alkhateeb, L. Rueda, W. ElMaraghy and A. Ngom (2018). Identification of the treatment survivability gene biomarkers of breast cancer patients via a tree-based approach. International Conference on Bioinformatics and Biomedical Engineering, Springer.

Tang, W., S. Wan, Z. Yang, A. E. Teschendorff and Q. Zou (2017). "Tumor origin detection with tissue-specific miRNA and DNA methylation markers." Bioinformatics 34(3): 398-406.

Tommasi, S., D. L. Karm, X. Wu, Y. Yen and G. P. Pfeifer (2009). "Methylation of homeobox genes is a frequent and early epigenetic event in breast cancer." Breast Cancer Research 11(1): R14.

Van De Vijver, M. J., Y. D. He, L. J. Van't Veer, H. Dai, A. A. Hart, D. W. Voskuil, G. J. Schreiber, J. L. Peterse, C. Roberts and M. J. Marton (2002). "A gene-expression signature as a predictor of survival in breast cancer." New England Journal of Medicine 347(25): 1999-2009.

Wang, K. (2007). "Applying data mining to manufacturing: the nature and implications." Journal of Intelligent Manufacturing 18(4): 487-495.

Wang, L., M. Törngren and M. Onori (2015). "Current status and advancement of cyber-physical systems in manufacturing." Journal of Manufacturing Systems 37(Part 2): 517-527.

Ward Jr, J. H. (1963). "Hierarchical grouping to optimize an objective function." Journal of the American statistical association 58(301): 236-244.

Widodo, A. and B.-S. Yang (2007). "Support vector machine in machine condition monitoring and fault diagnosis." Mechanical systems and signal processing 21(6): 2560-2574.

Wu, D., D. W. Rosen, L. Wang and D. Schaefer (2015). "Cloud-based design and manufacturing: A new paradigm in digital manufacturing and design innovation." Computer-Aided Design 59: 1-14.

Xu, L. D., E. L. Xu and L. Li (2018). "Industry 4.0: state of the art and future trends." International Journal of Production Research 56(8): 2941-2962.

Xu, X. (2012). "From cloud computing to cloud manufacturing." Robotics and computer-integrated manufacturing 28(1): 75-86.

Yu, Z., Y. Lu, J. Zhang, J. You, H. Wong, Y. Wang and G. Han (2018). "Progressive Semisupervised Learning of Multiple Classifiers." IEEE Transactions on Cybernetics 48(2): 689-702.

Zachman, J. A. (1987). "A framework for information systems architecture." IBM systems journal 26(3): 276-292.

Zaharia, M., R. S. Xin, P. Wendell, T. Das, M. Armbrust, A. Dave, X. Meng, J. Rosen, S. Venkataraman and M. J. Franklin (2016). "Apache spark: a unified engine for big data processing." Communications of the ACM 59(11): 56-65.

Zeng, X., L. Liu, L. Lü, Q. Zou and A. Valencia (2018). "Prediction of potential disease-associated microRNAs using structural perturbation method." Bioinformatics 1: 8.

Zhang, J. J., F.-Y. Wang, X. Wang, G. Xiong, F. Zhu, Y. Lv, J. Hou, S. Han, Y. Yuan and Q. Lu (2018). "Cyber-physical-social systems: The state of the art and perspectives." IEEE Transactions on Computational Social Systems 5(3): 829-840.

Zhao, R., R. Yan, Z. Chen, K. Mao, P. Wang and R. X. Gao (2016). "Deep learning and its applications to machine health monitoring: A survey." arXiv preprint arXiv:1612.07640.

Zheng, Z., J. Zhu and M. R. Lyu (2013). Service-generated big data and big data-as-a-service: an overview. 2013 IEEE international congress on Big Data, IEEE.

Zhong, R. Y., G. Q. Huang, S. Lan, Q. Dai, X. Chen and T. Zhang (2015). "A big data approach for logistics trajectory discovery from RFID-enabled production data." International Journal of Production Economics 165: 260-272.

Zou, Q., J. Zeng, L. Cao and R. Ji (2016). "A novel features ranking metric with application to scalable visual and bioinformatics data classification." Neurocomputing 173: 346-354.

APPENDIX A: SAMPLE OF PATIENTS GENE EXPRESSION AND CLINICAL DATA

Around 25,000 genes expression for almost 2,433 patients which are about 60,000,000 gene expression values.

Table 16. A sample of the patient's clinical data.

	Clinical Data						
	MONTHS	**STATUS**	**SURGERY**	**HORMONE THERAPY**	**….**	**RADIO THERAPY**	**CHEMO THERAPY**
Patient 1	140.5	Living	MASTECTOMY	YES	….	YES	NO
Patient 2	84.63	Living	BREAST CONSERVING	YES	….	YES	NO
Patient 3	163.1	Died of Disease	MASTECTOMY	YES	….	NO	YES
Patient 4	164.93	Living	MASTECTOMY	YES	….	YES	YES
Patient 5	41.37	Died of Disease	MASTECTOMY	YES	….	YES	YES
Patient 6	7.8	Died of Disease	MASTECTOMY	YES	….	YES	NO
Patient 7	164.33	Living	BREAST CONSERVING	YES	….	YES	YES
Patient 8	22.4	Died of Disease	MASTECTOMY	NO	….	YES	YES
Patient 9	99.53	Died of Other Causes	BREAST CONSERVING	YES	….	YES	NO
Patient 10	36.57	Died of Other Causes	BREAST CONSERVING	YES	….	YES	NO
Patient 11	36.27	Died of Disease	MASTECTOMY	NO	….	NO	NO
Patient 12	132.03	Died of Disease	MASTECTOMY	YES	….	YES	NO
Patient 13	163.53	Living	BREAST CONSERVING	YES	….	YES	NO
Patient 14	164.9	Living	MASTECTOMY	NO	….	YES	YES
Patient 15	14.13	Died of Other Causes	MASTECTOMY	YES	….	YES	NO
….	….	….	….	…..	….	….	….

Table 17. A sample of the patient's genes expression with desired classes.

	Gene Expression						
	LIN52	**PCOTH**	**GRM1**	**SLC9A1**	**….**	**CD164**	**Class**
Patient 1	6.601455	6.615954	5.370035	9.356541	….	7.222594	**DR**
Patient 2	6.006671	5.640276	5.322082	9.364857	….	6.523877	**LR**
Patient 3	7.263404	5.807551	5.418732	11.19858	….	6.687907	**LS**
Patient 4	6.323518	5.455436	5.51028	9.199171	….	5.688272	**LS**
Patient 5	6.409583	5.572757	5.199104	9.334389	….	5.845664	**DS**
Patient 6	6.414613	5.721091	5.061777	9.529512	….	5.870183	**DH**
Patient 7	6.563254	5.449545	5.380782	8.887233	….	5.712453	**DS**
Patient 8	6.812145	5.645348	5.287389	8.90297	….	5.751436	**LH**
Patient 9	6.323518	5.455436	5.51028	9.199171	….	5.688272	**LS**
Patient 10	6.409583	5.572757	5.199104	9.334389	….	5.845664	**DS**
Patient 11	6.414613	5.721091	5.061777	9.529512	….	5.870183	**DH**
Patient 12	6.563254	5.449545	5.380782	8.887233	….	5.712453	**DS**
Patient 13	6.812145	5.645348	5.287389	8.90297	….	5.751436	**LH**
Patient 14	6.657678	5.635089	5.291151	9.858843	….	6.44788	**LS**
Patient 15	5.961513	5.551212	5.153493	9.508277	….	5.802332	**DS**
….	….	….	….	…..	….	….	**….**

APPENDIX B: COMPUTE CANADA SETUP AND CNN CODE FOR MANUFACTURING SYSTEM (SAMPLE)

A. Compute Canada Setup (Sample)

```
module load python/3.6
virtualenv $HOME/jupyter_py3
source $HOME/jupyter_py3/bin/activate
pip install jupyter
echo -e '#!/bin/bash\nunset XDG_RUNTIME_DIR\njupyter notebook --ip $(hostname -f) --no-browser' > $VIRTUAL_ENV/bin/notebook.sh
chmod u+x $VIRTUAL_ENV/bin/notebook.sh
pip install jupyterlmod
jupyter nbextension install --py jupyterlmod --sys-prefix
jupyter nbextension enable --py jupyterlmod --sys-prefix
jupyter serverextension enable --py jupyterlmod --sys-prefix
salloc --time=8:0:0 --ntasks=4 --nodes=4 --cpus-per-task=16 --mem-per-cpu=100G --account=def-Ashraf srun $VIRTUAL_ENV/bin/notebook.sh
```

B. CNN Code (Sample)

```
!pip install opencv-python
#!pip install opencv2
!pip install keras
!pip install gast
!pip install --upgrade tflearn
#!pip install tensorflow_cpu
!pip install tflearn
!pip install SinchSMS


import cv2                    # working with, mainly resizing, images
```

```
import numpy as np                  # dealing with arrays
import os                           # dealing with directories
from random import, shuffle         # mixing up or currently ordered data that might lead our network
astray in training.
from tqdm import tqdm               # a nice pretty percentage bar for tasks. Thanks to viewer Daniel
BA1/4hler for this suggestion

!pip install tqdm

#TRAIN_DIR = 'train_n'
GOOD_DIR = 'good_ash'
BAD_DIR = 'bad_ash'
TEST_DIR = 'final_test'
IMG_SIZE = 512
LR = 0.001

MODEL_NAME = 'Ifact-()-().model'.format(LR, '2conv-basic') # just so we remember which saved
model is which, sizes must match

def create_train_data():
    training_data = []
    for img in tqdm(os.listdir(GOOD_DIR)):
        label = [1.0,0.0] # 'good'
        path = os.path.join(GOOD_DIR,img)
        img = cv2.imread(path,cv2.IMREAD_GRAYSCALE)
        img = cv2.resize(img, (IMG_SIZE,IMG_SIZE))/255
        training_data.append([np.array(img),np.array(label)])

    for img in tqdm(os.listdir(BAD_DIR)):
        label = [0.0,1.0] #'bad'
        path = os.path.join(BAD_DIR,img)
        img = cv2.imread(path,cv2.IMREAD_GRAYSCALE)
        img = cv2.resize(img, (IMG_SIZE,IMG_SIZE))/255
        training_data.append([np.array(img),np.array(label)])
    shuffle(training_data)
    np.save('train_data.npy', training_data)
    return training_data
```

```
def process_test_data():
    testing_data = []
    for img in tqdm(os.listdir(TEST_DIR)):
        path = os.path.join(TEST_DIR,img)
        img_num = img.split('.')[0]
        img = cv2.imread(path,cv2.IMREAD_GRAYSCALE)
        img = cv2.resize(img, (IMG_SIZE,IMG_SIZE))
        testing_data.append([np.array(img), img_num])
    shuffle(testing_data)
    np.save('test_data.npy', testing_data)

train_data = create_train_data()
test_data = process_test_data()
# If you have already created the dataset:
#train_data = np.load('train_data.npy')

!pip install tflearn
!pip install tensorflow_cpu
!pip install --upgrade tflearn

krl = 3
drp_out_per = 0.2
epoc_n = 10
bsize = 60

import tensorflow as tf
import tflearn
from tflearn.layers.conv import conv_2d, max_pool_2d
from tflearn.layers.core import input_data, dropout, fully_connected
from tflearn.layers.estimator import regression

tf.reset_default_graph()

convnet = input_data(shape=[None, IMG_SIZE, IMG_SIZE, 1], name='input')
convnet = conv_2d(convnet, 32, 5, activation='relu')
convnet = max_pool_2d(convnet, 5)
#convnet = dropout(convnet, drp_out_per)
```

```
convnet = conv_2d(convnet, 64, 5, activation='relu')
convnet = max_pool_2d(convnet, 5)
#convnet = dropout(convnet, drp_out_per)

convnet = conv_2d(convnet, 128, 5, activation='relu')
convnet = max_pool_2d(convnet, 5)
#convnet = dropout(convnet, drp_out_per)
convnet = conv_2d(convnet, 256, 5, activation='relu')
convnet = max_pool_2d(convnet, 5)
#convnet = dropout(convnet, drp_out_per)

convnet = conv_2d(convnet, 512, 5, activation='relu')
convnet = max_pool_2d(convnet, 5)
#convnet = dropout(convnet, drp_out_per)

convnet = fully_connected(convnet, 1024, activation='relu')
convnet = dropout(convnet, drp_out_per)

convnet = fully_connected(convnet, 2, activation='softmax')
convnet = regression(convnet, optimizer='adam', learning_rate=LR, loss='categorical_crossentropy',
name='targets')

model = tflearn.DNN(convnet, tensorboard_dir='log')

if os.path.exists('().meta'.format(MODEL_NAME)):
   model.load(MODEL_NAME)
   print('model loaded!')
train = train_data[:-30]
test = train_data[-30:]
X = np.array([i[0] for i in train]).reshape(-1,IMG_SIZE,IMG_SIZE,1)
Y = [i[1] for i in train]
test_x = np.array([i[0] for i in test]).reshape(-1,IMG_SIZE,IMG_SIZE,1)
test_y = [i[1] for i in test]

model.fit(('input': X), ('targets': Y), n_epoch=epoc_n, batch_size=bsize ,
        validation_set=(('input': test_x), ('targets': test_y)),
        snapshot_step=500, show_metric=True, run_id=MODEL_NAME)
```

C. CNN Testing Code (Sample)

```
X_DIR = 'X'
pic = 'X\\IMG_0225.JPG'
import numpy as np
from PIL import Image
import matplotlib.pyplot as plt
%matplotlib inline
def create_X_data(var):
    training_data = []
    label = [0.0,1.0] #'bad'
    img = cv2.imread(var,cv2.IMREAD_GRAYSCALE)
    img = cv2.resize(img, (IMG_SIZE,IMG_SIZE))/255
    training_data.append([np.array(img),np.array(label)])

    shuffle(training_data)
    np.save('train_data.npy', training_data)

    return training_data


#########

X_data = create_X_data(pic)
print(len(X_data))
X_1 = []
X_1 = np.array([i[0] for i in X_data]).reshape(-1,IMG_SIZE,IMG_SIZE,1)
Y = [i[1] for i in X_data]
#########

proba = model.predict(X_1)

idxs = np.argsort(proba)[::-1][:2]
y = ""
if idxs[0][1]==1:
    y="This is a Bad Product"
```

```
else:
    y="This is a Good Product"

image_path = cv2.imread(pic)

plt.imshow(image_path)
plt.title(y.upper())
plt.show()

###########

# Sending SMS to a phone number

import time
from sinchsms import SinchSMS

number = '+1519300xxxx'

message = y

#client = SinchSMS('e241b001-5856-453d-8fdf-c6fb8cfb8ef1', 'kO5IPW8jA0ORnYBoTbG6WA==')
client = SinchSMS('ce00e956-0ec9-47f7-9000-e68a6926a964', 'mN1Udit8NkeRdcpHSOb3mQ==') #
Ashraf
#client = SinchSMS('afc7e725-edeb-4d48-a2bb-5276c1bf118d', 'kGbF27vt/E6vNDMnNytqFA==') #
Dr Waguih
print("Sending '%s' to %s" % (message, number))
response = client.send_message(number, message)
message_id = response['messageId']

response = client.check_status(message_id)
while response['status'] != 'Successful':
    print(response['status'])
    time.sleep(1)
    response = client.check_status(message_id)
    print(response['status'])
```

D. CNN Web Cam Code (Sample)

```
import cv2

cap=cv2.VideoCapture(0)
if cap.isOpened():
   ret, frame=cap.read()
   print(ret)
   #print(frame)
else:
   ret=False



##########



imge1=cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY  ) #cv2.COLOR_BGR2GRAY)
plt.imshow(imge1)
plt.title("color image ")
plt.xticks([])
plt.yticks([])
plt.show()

# save image
status = cv2.imwrite('testImage.png',imge1)

## If already ran please release the camera
cap.release()
del (cap)

########
########

pic = 'testImage.png'
X_data = create_X_data(pic)
print(len(X_data))
```

```
X_1 = []
X_1 = np.array([i[0] for i in X_data]).reshape(-1,IMG_SIZE,IMG_SIZE,1)
Y = [i[1] for i in X_data]


#######

proba = model.predict(X_1)

idxs = np.argsort(proba)[::-1][:2]

y = ""
if idxs[0][1]==1:
   y="This is a Bad Product"
else:
   y="This is a Good Product"

image_path = cv2.imread(pic)

plt.imshow(image_path)
plt.title(y.upper())
plt.show()


# Sending SMS to a phone number

import time
from sinchsms import SinchSMS

# Mostafa phone number

number = '+1519300xxxx'
message = y


client = SinchSMS('ce00e956-0ec9-47f7-9000-e68a6926a964', 'mN1Udit8NkeRdcpHSOb3mQ==')
```

```
print("Sending '%s' to %s" % (message, number))
response = client.send_message(number, message)
message_id = response['messageId']

response = client.check_status(message_id)
while response['status'] != 'Successful':
    print(response['status'])
    time.sleep(1)
    response = client.check_status(message_id)
    print(response['status'])
```

E. CNN Code Results (Sample)

```
Training Step: 1659  | total loss: 0.17452 | time: 46.930s
| Adam | epoch: 010 | loss: 0.17452 - acc: 0.9451 -- iter: 9940/10000
Training Step: 1660  | total loss: 0.16441 | time: 53.279s
| Adam | epoch: 010 | loss: 0.16441 - acc: 0.9671 | val_loss: 0.10440 - val_acc: 1.0000 -- iter:
10000/10000
```

APPENDIX C: CLASS IMBALANCE TECHNIQUES FOR HEALTH INFORMATICS SYSTEM (SAMPLE)

Class Imbalance

The heath informatics models uses a one-versus-rest scheme to tackle the multi-class problem, which leads to an imbalanced class dataset at each node of the classification model. Therefore, we applied the following techniques to handle this issue:

A. Over-sampling with synthetic data

Oversampling the minority class by using synthetic data generators. Several algorithms are used to achieve this. We used one of the most popular ones, SMOTE.

For example the run file for DH Vs Rest samples at node number five (N5) at case study number three for the single linkage shows:

Scheme: weka.classifiers.trees.RandomForest -P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1

Relation: final_gene_5Y_class_347_DH_Rest-weka.filters.unsupervised.attribute.Remove-V-R7175,23955,4653,7209,4676,3980,24369-weka.filters.unsupervised.attribute.Remove-V-R1-10,21-weka.filters.supervised.instance.SMOTE-C0-K5-P517.0-S1

Test mode: 10-fold cross-validation

B. Using a cost-sensitive classifier

Using penalizing models that apply additional weight to the minority class to achieve class balancing. This, in turn, biases the model to pay more attention to the minority class than others. The algorithm utilized in this work is called Cost-Sensitive Classifier in the Weka machine learning tool using a penalty matrix to overcome the imbalance.

For example the run file for DHLHLR Vs DRDSLS samples at node number five (N5) at case study number three for the ward linkage shows:

Scheme: weka.classifiers.meta.CostSensitiveClassifier -cost-matrix "[0.0 1.0; 1.1111 0.0]" -S 1 -W weka.classifiers.trees.RandomForest -- -P 100 -I 100 -num-slots 1 -K 1 -M 1.0 -V 0.001 -S 1

Relation: 5_final_gene_5Y_class_347_DHLHLR_DRDSLS-weka.filters.unsupervised.attribute.Remove-V-,785,1132, weka.filters.unsupervised.attribute.Remove-V-R2,8,12-13,15,17,36,48,67,75,2243

Test mode: 10-fold cross-validation

Cost Matrix

0 1

1.11 0

C. Resampling

Replicating the dataset can be using one of two methods: (1) adding copies of the data instances to the minority class, which is called over-sampling (2) deleting some instances of the majority class, which is called under-sampling. We used the over-sampling technique.

For example the run file for DH Vs LH samples at node number one (N1) at case study number three for the ward linkage shows:

Scheme: weka.classifiers.trees.RandomForest -P 100 -I 100 -num-slots 1 -K 1 -M 1.0 -V 0.001 -S 1

Relation: 1_final_gene_5Y_class_26_DH_LH_503_2_NB_2X

Instances: 32

Attributes: 3

INO80

PAX7

class

Test mode: 10-fold cross-validation

VITA AUCTORIS

NAME: Ashraf Abou Tabl

PLACE OF BIRTH: Egypt

YEAR OF BIRTH: 1973

EDUCATION: B.Sc. in Bachelor of Computer Engineering, Benha University, 1996.
M.Sc. in Computer science and Engineering, Menofyia University, Egypt, 2008.

PUBLICATIONS: **Journal Papers**

1. H.M.Kelash , Ashraf M Abou Tabl, 2007, New Architecture of Interconnect for High-Speed Optical Computerized Data Networks (Nonlinear Response), Menoufia Journal of Electronic Engineering Research (MJEER), Volume 17, Number 2, pages 199-216 .
2. El-Badawy, El-Sayed A.; EI-Halafawy, Farag Z.; Kelash, Hamdy A.; Abou-Tabl, Ashraf M. , 2007, New Architecture of Optical Interconnect for High-Speed Optical Computerized Data Networks (Nonlinear Response), WMSCI 2007 - The 11th World MultiConference on Systemics, Cybernetics and Informatics, Jointly with the 13th International Conference on Information Systems Analysis and Synthesis, ISAS 2007 - Proc., volume 6 - number 1 - year 2008, pages 73-79.
   http://www.iiisci.org/journal/sci/Abstract.asp?var=&id=S069CS
   HTTP://www.iiisci.org/journal/CV$/sci/pdfs/S069CS.pdf
3. Ashraf Abou Tabl, A. Alkhateeb, P. Quang, L. Rueda, W. ElMaraghy, A. Ngom.” A novel approach for identifying relevant genes for breast cancer survivability on specific therapies” Evolutionary Bioinformatics, 2018, Evolutionary Bioinformatics, 14, doi: 10.1177/1176934318790266.
4. Ashraf Abou Tabl, A. Alkhateeb, L. Rueda, W. ElMaraghy, A. Ngom. “A machine learning approach for identifying gene biomarkers guiding the treatment of breast cancer,” Frontiers in Genetics. doi: 10.3389/fgene.2019.00256.

5. Ashraf Abou Tabl, W. ElMaraghy. "Big Data Analytics for Defects Detection in Manufacturing Systems (Industry 4.0)", International Journal of Computer Integrated Manufacturing(IJCIM) ID: TCIM-2019-IJCIM-0178. [Submitted]
6. S. Jubair, A. Alkhateeb, A Abou Tabl, L. Rueda, A. Ngom, "Identifying subtype specific network-biomarkers of breast cancer survivability", Evolutionary Bioinformatics, (EVB-2019-0142), [Submitted].
7. Abou Tabl A, Alkhateeb A, ElMaraghy W and Ngom A. Machine learning model for identifying gene biomarkers for breast cancer treatment survival [version 1; not peer reviewed]. F1000Research 2017, 6(ISCB Comm J):1681 (doi: 10.7490/f1000research.1114873.1) (Non-referee Journal) (Abstract and Poster)

Conference Papers and Posters

8. Abou Tabl, A., Alkhateeb, A., ElMaraghy, W., & Ngom, A. (2017, August). Machine Learning Model for Identifying Gene Biomarkers for Breast Cancer Treatment Survival. In Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics (pp. 607-607) (ACM 2017), Boston, MA. doi>10.1145/3107411.3108217
9. Abou Tabl, A., Alkhateeb, A., ElMaraghy, W., & Ngom, A. (2017, August). Machine Learning Model for Identifying Gene Biomarkers for Breast Cancer Treatment Survival. In Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics. ACM (Poster)
10. Abou Tabl, A., Alkhateeb, A., Rueda, L., ElMaraghy, W., & Ngom, A. (2018, March). Identifying gene biomarkers for breast cancer survival using a tree-based approach. BHI 2018 IEEE International Conference on Biomedical and Health Informatics, Las Vegas, NV, USA, March 2018.
11. Ashraf Abou Tabl, A. Alkhateeb, L. Rueda, W. ElMaraghy, A. Ngom. "Identification of the Treatment Survivability Gene Biomarkers of Breast Cancer Patients via a Tree-Based Approach." 6th International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO 2018), Granada, Spain, 2018. (pp. 166-176). Springer, Cham.
12. Ashraf M Abou Tabl, ElMaraghy, W., 2016, ``Big Data Management for Sustainable Urban Transportation``, University of Windsor, Faculty of Engineering, Poster at UWindsor Urban Transportation Event, November 11, 2016.

13. Abou Tabl, A., Moussa, M., ElMaraghy, W., ElMaraghy, H., Emerging Technologies in Automation Conference and Trade Show hosted by Windsor Essex Economic Development Corporation's (WEEDC). Oct. 31, 2017, Windsor, Canada. (Poster).